

# Hybride Simulation thermischer Systeme am Beispiel eines Reisebusses

Von der Fakultät für Maschinenbau  
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung der Würde eines  
Doktor - Ingenieurs (Dr. - Ing.)

genehmigte Dissertation

von: Roland Kossel  
aus (Geburtsort): Osnabrück

Referenten: Prof. Dr. - Ing. Jürgen Köhler  
Prof. Dr. - Ing. Gerhard Schmitz  
Vorsitzender: Prof. Dr. - Ing. Ferit Küçükay.

Eingereicht am: 03. März 2011  
Mündliche Prüfung am: 31. Oktober 2011



# Danksagung

Die vorliegende Dissertation konnte ich als Mitarbeiter der TLK-Thermo GmbH in Braunschweig in enger Zusammenarbeit mit dem Institut für Thermodynamik der Technischen Universität Braunschweig erarbeiten. Hierbei wurde es mir ermöglicht, das angeeignete Wissen in Industrieprojekten anzuwenden und dieses darüber hinaus wissenschaftlich weiterzuentwickeln. Dass mir die hierfür benötigte Zeit eingeräumt wurde, rechne ich der TLK-Thermo GmbH hoch an. Der Dank hierfür gilt den Geschäftsführern Dr. - Ing. Wilhelm Tegethoff und Dr. - Ing. Nicholas Lemke sowie den Kollegen, die mir vor allem in der Endphase den Rücken freigehalten haben.

Herrn Prof. Dr. - Ing. Jürgen Köhler vom Institut für Thermodynamik der Technischen Universität Braunschweig und Herrn Prof. Dr. - Ing. Gerhard Schmitz vom Institut für Thermofluid-dynamik der Technischen Universität Hamburg-Harburg danke ich für die Übernahme des Erst- bzw. Zweitgutachtens sowie die fachlichen und privaten Diskussionen und Gespräche. Dank geht auch an Herrn Prof. Dr. - Ing. Ferit Küçükay für die Übernahme des Vorsitzes der Prüfungskommission.

Herrn Dr. - Ing. Wilhelm Tegethoff gebührt besonderer Dank für die fachliche Betreuung seitens der TLK-Thermo GmbH. In den zahllosen Diskussionen, die wir geführt haben, sind viele Ideen entstanden, gereift, ausgearbeitet und gelegentlich auch verworfen worden. Für auftretende Probleme und Fragen hatte er stets ein offenes Ohr.

In meinem Bürokollegen Christian Strupp habe ich in den vergangenen Jahren nicht nur einen überaus kompetenten Ansprech- und Diskussionspartner für jedwede Fragestellung gehabt. Über das Fachliche hinaus ist zwischen uns eine freundschaftliche Bindung entstanden, die ich nicht mehr missen möchte. Für alles: Vielen Dank. Seiner Frau Julia Kunze danke ich dafür, dass mir Christian so oft helfen durfte. Für die zahllosen Koch-, Film-, Spiele- und Gesprächsabende mit Julia und Christian kann ich nicht genug Danke sagen.

Viele weitere Menschen in meinem Umfeld haben – bewusst und unbewusst – daran mitgewirkt, dass die vorliegende Arbeit zustande kommen konnte. Da ich hier leider nicht jeden namentlich erwähnen kann, zum Abschluss ein allgemeines Dankeschön an diese Personen.

Braunschweig im Januar 2011

Roland Kossel





# Inhaltsverzeichnis

<b>1. Einleitung</b>	1
1.1. Stand der Wissenschaft	4
1.2. Ziele der Arbeit	7
1.3. Aufbau der Arbeit	8
<b>2. Ausgewählte Aspekte der Kopplung</b>	11
2.1. Klassifizierung von Modellbildung und Kopplung	11
2.2. Synchronisation	14
2.2.1. Parallele Synchronisation	16
2.2.2. Serielle Synchronisation	17
2.2.3. Synchronisationsschrittweite	18
2.3. Kommunikation im Programm-/Prozessverbund	21
2.4. Auswirkung der Kopplung auf die Simulationszeit	23
<b>3. Behandlung Numerischer Effekte bei der Kopplung</b>	27
3.1. Glättung und Extrapolation empfangener Werte	28
3.1.1. Glättung	29
3.1.2. Extrapolation	31
3.2. Erhaltung der vollen Dynamik von Randwerten	34
3.3. Auswahl und Behandlung auszutauschender Variablen	37
3.4. Kopplungseffekte bei Systemsimulationen	41
3.4.1. Betrachtetes System	41
3.4.2. Simulation stationärer und transienter Randbedingungen	47
3.4.3. Auswirkungen auf Entwicklungs- und Rechenzeit	49
<b>4. Umsetzung der Konzepte</b>	53
4.1. Kommunikation während der Simulation	54
4.2. Zentrale Ansteuerung der Simulationsprogramme	56
4.3. Eingesetzte Übertragungstechnik	59
<b>5. Arbeitsprozess in Unternehmen</b>	61
<b>6. Simulation des thermischen Gesamtsystems eines Reisebusses</b>	65
6.1. Aufteilung in Teilsysteme	66

6.2. Modelle der Teilsysteme . . . . .	69
6.2.1. Kältekreislauf . . . . .	69
6.2.2. Kühl-/Heizkreislauf . . . . .	70
6.2.3. Innenraum . . . . .	71
6.2.4. Längsdynamik . . . . .	72
6.3. Bestimmung der Kosimulationseinstellungen . . . . .	73
6.3.1. Ausrollen . . . . .	74
6.3.2. Tunnelfahrt . . . . .	76
6.4. Reise Lissabon – Madrid . . . . .	80
<b>7. Zusammenfassung . . . . .</b>	<b>93</b>
<b>A. Literaturverzeichnis . . . . .</b>	<b>97</b>
<b>B. Eigene Veröffentlichungen . . . . .</b>	<b>111</b>

# 1. Einleitung

In der Entwicklung technischer Systeme nimmt die Computersimulation seit langem eine Schlüsselposition ein. Neben Experimenten an physischen Prototypen ist die Simulation oft die einzige Möglichkeit, das Betriebsverhalten zukünftiger Systeme vorherzusagen [Cellier u. a. 1996]. Durch ihren Einsatz kann nicht nur die Entwicklungszeit technischer Systeme wesentlich verkürzt werden, es wird auch ermöglicht, potentielle Probleme frühzeitig zu entdecken und auf diese zu reagieren. Dies ist besonders wichtig, da Fehler, die erst in der Erprobungsphase entdeckt werden, immense Kosten verursachen [Puntigam u.a. 2005a, Dreier 2006].

Im Front-Loading – dem Ziel, das Verhalten eines Produktes durch virtuelle Methoden so früh wie möglich im Entwicklungsprozess abbilden und untersuchen zu können – nimmt die Simulation die wichtigste Stelle ein [Thomke 2003]. Wie in Abbildung 1.1 dargestellt, werden durch den Einsatz von Simulation schnelle Varianten-Iterationen ermöglicht (1). In Kombination mit der in früheren Projekten gesammelten Erfahrung (2) kann das Produkt mit gleicher Qualität wesentlich schneller und somit kostengünstiger zur Marktreife gebracht werden. Gleichzeitig erlaubt die Simulation, die durchgeführten Versuche besser zu planen, wodurch breiter angelegte Experimente möglich sind (3). Das Ergebnis dieser drei Punkte ist eine Verlagerung des Entwicklungsverlaufs nach links oben, also hin zu einer kürzeren Entwicklungszeit bei höherer Produktqualität.

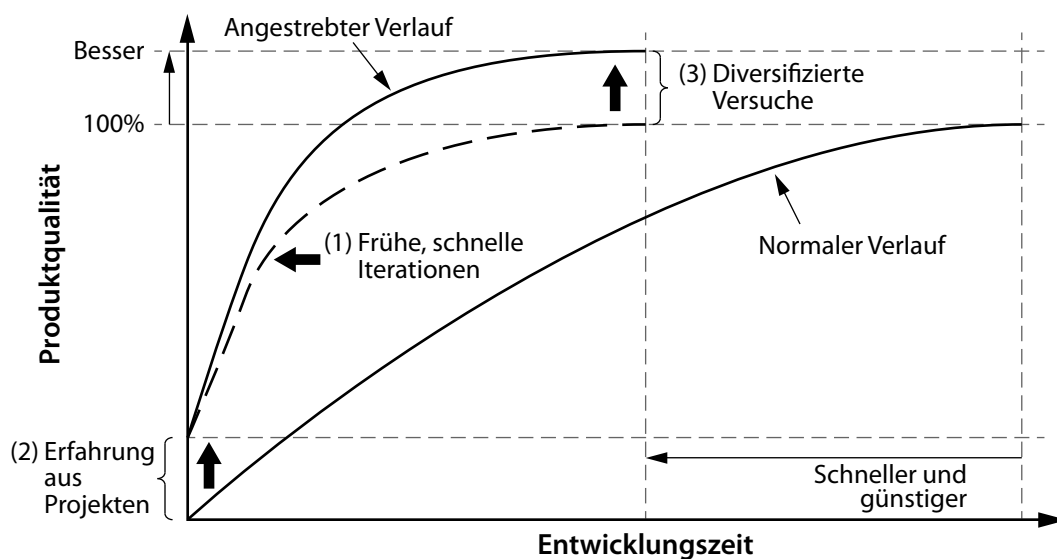


Abbildung 1.1.: Front-Loading des Entwicklungsverlaufs durch den Einsatz virtueller Techniken [nach Thomke 2003].

---

Besteht das zu untersuchende System aus Teilen, die unterschiedlichen technischen Disziplinen zuzuordnen sind, lassen sich diese häufig nicht mit dem gewünschten Detaillierungsgrad innerhalb eines Programms modellieren. In einem solchen Fall erlaubt es die hybride Simulation, mehrere Simulationswerkzeuge zu koppeln; mehr hierzu in den folgenden Absätzen.

Die Auslegung der einzelnen Teile eines Gesamtsystems erfolgt klassischer Weise in einzelnen, voneinander getrennten Simulationen unter stationären Randbedingungen [Lund u. a. 2005]. Für diese Auslegung der einzelnen Teilsysteme sind in der Industrie die entsprechenden Fachabteilungen zuständig, die für die Modellierung und Simulation spezialisierte Simulationsprogramme verwenden. Da solche Programme oft ausschließlich den zu untersuchenden Teilaspekt abbilden können oder nur rudimentäre Modelle für die angrenzenden Systeme enthalten, können Gesamtsysteme mit diesen Werkzeugen nicht oder nur unzureichend abgebildet werden. Die Vielfalt der möglichen Programme wird in Trčka [2008] deutlich, wo allein die Simulation von Heizungs- und Klimasystemen in sieben Bereiche aufgeteilt und für jeden dieser Bereiche mehrere Beispiele für Werkzeuge angegeben werden. Da heute das Hauptpotential für weitergehende Optimierungen in der detaillierten Abstimmung des Gesamtsystemverhaltens bei transienten Randbedingungen liegt [Hager u. a. 2003], muss somit die bisherige Arbeitsweise erweitert werden [Betz u. a. 2005].

Um die Einzelteile bereits vor dem Bau eines physischen Prototyps funktional auf einander abzustimmen [Binder u. a. 2008] und Auswirkungen baulicher Veränderungen prognostizieren zu können [Betz u. a. 2005, Lund u. a. 2005], ist die Simulation des Gesamtsystems als virtueller Prototyp notwendig. Hierbei soll eine Simulationsgüte erreicht werden, die das Einsparen ganzer Baureihen physischer Prototypen ermöglicht. Dieses Ziel wird teilweise bereits erreicht und ist Gegenstand derzeitiger Forschung [Baron 2008, Maaßen u. Pischinger 2008]. Die Aufgabe der mit den physischen Prototypen durchgeführten Tests verlagert sich hin zur Validierung der virtuellen Prototypen [Seiffert u. Rainer 2008]. Bei einfachen zu untersuchenden Systemen kann ein solcher Prototyp durch eine Kette von Simulationswerkzeugen aufgebaut werden, in der die Ergebnisse eines Modells die Eingabewerte des nachgelagerten Modells darstellen. Da bei diesem Vorgehen keine Informationen zum vorgelagerten Modell übergeben werden können, werden die Rückwirkungen vernachlässigt. Um komplexe Gesamtsysteme detailliert abbilden und untersuchen zu können, müssen jedoch sämtliche Wechselwirkungen zwischen allen Teilsystemen berücksichtigt werden [Puntigam u. a. 2006a, Schröder u. a. 2002]. So ist beispielsweise eine Optimierung der Klimaanlage nur dann sinnvoll, wenn die Wechselwirkungen mit den übrigen Komponenten berücksichtigt werden: Jones u. Fletcher [2005] untersuchten Ölkühler, Ladeluftkühler, Verflüssiger und Hauptwasserkühler, die im Fahrzeug als Paket angeordnet werden. Im Vergleich zur freien Anströmung weist der Hauptwasserkühler hierbei einen Leistungsverlust von bis zu 29,3% auf – jede Änderung an einer der vorgeschalteten Komponenten beeinflusst somit auch alle nachfolgenden Bauteile.

Zur Erstellung eines virtuellen Prototypen für die Simulation eines Gesamtsystems unter Berücksichtigung der Wechselwirkungen zwischen den Teilsystemen stehen nach Tegethoff u. a. [2006a] zwei Wege zur Auswahl: (1) Das Neuerstellen des Gesamtsystemmodells in einer ge-

eigneten Modellierungssprache und (2) die Kopplung der mit verschiedenen Simulationsprogrammen erzeugten Modelle in einer Kosimulation.

Weg (1) – das Neuerstellen – widerspricht der Maxime, bestehende Modelle weiterzuverwenden. Die Möglichkeiten, bestehende Modelle weiterzuverwenden, lassen sich nach Pidd [2002] in vier Kategorien, vom kleinen Codestück bis zum kompletten Modell, einteilen (siehe Abbildung 1.2). Die einfachste und deshalb auch am häufigsten eingesetzte Methode ist das Wiederverwenden von Quellcodeausschnitten; das Kopieren spezialisierter Funktionen wie z.B. Wärmeübergangsbeziehungen ist der nächste Schritt. Während die Einbindung von Komponenten – einzelner Teile des Gesamtmodells – bereits eine Herausforderung darstellt, ist das Wiederverwenden des Gesamtmodells die Königsdisziplin.

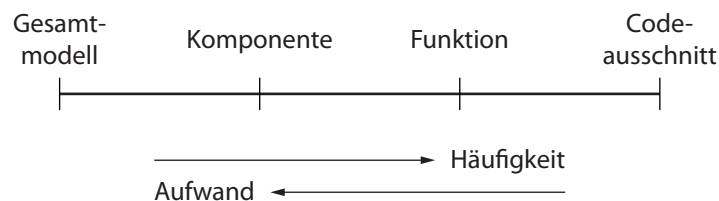


Abbildung 1.2.: Möglichkeiten der Wiederverwendung von Modellen und Modellbestandteilen [nach Pidd 2002].

In einer in Robinson u. a. [2004] beschriebenen Podiumsdiskussion sehen die teilnehmenden Experten bezüglich des Wiederverwendens bestehender Modelle das Problem, die Modellanpassung zu finanzieren, da die Kosten bei der Modellentwicklung entstehen, den Nutzen aber der Anwender hat und dieser üblicherweise einer anderen Organisationseinheit angehört. Es sei jedoch trotz aller Probleme nicht nur sinnvoll, sondern wirtschaftlich notwendig, bestehenden Simulationscode wiederzuverwenden.

Die hybride Simulation – das Verbinden verschiedener Simulationstechniken in einem Verbund – stellt eine Möglichkeit dar, den oben beschriebenen Weg (2) umzusetzen: Bestehende Modelle potentiell unterschiedlicher Simulationsprogramme werden weiterverwendet und im Rahmen einer Kosimulation zu einem virtuellen Prototypen des Gesamtsystems zusammengeschlossen. Nach Lund u. a. [2008] müssen beim Aufbau der Kosimulation zunächst „lästige aber wichtige formale Hürden“ überwunden werden, wie z.B. das Festlegen von Namen, Einheiten und Vorzeichen der ausgetauschten Variablen sowie der Kopplungsstruktur des Gesamtsystems, d.h. die Partitionierung des Gesamtsystems in Teilsysteme (siehe Abbildung 1.3). Komponenten, die in mehreren Modellen abgebildet werden (Wärmeübertrager sind hierfür ein typisches Beispiel), benötigen eine tiefergehende Betrachtung. Auch wenn für den Aufbau einer solchen Kosimulation Anpassungen an den Modellen vorgenommen werden müssen, um klar definierte Schnittstellen zu schaffen, können sie weitestgehend unverändert wiederverwendet werden.

Für die hybride Simulation spricht über die Weiterverwendung bestehender Modelle hinausgehend, dass die einzelnen Modelle einfach ausgetauscht werden können und der Detaillierungs-

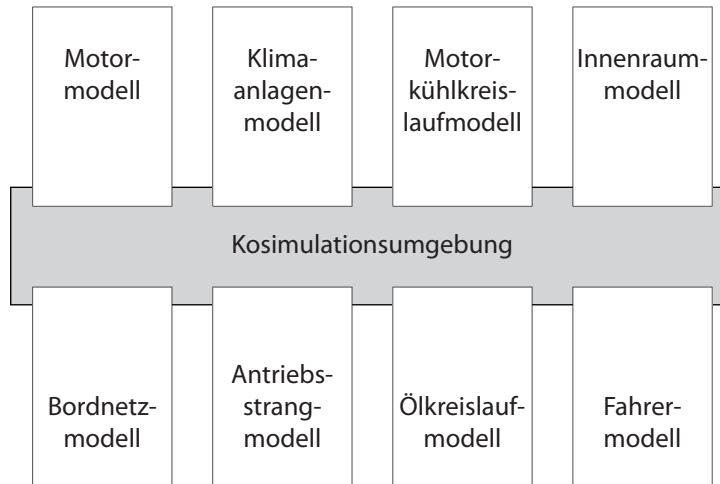


Abbildung 1.3.: Kopplungsstruktur eines Gesamtfahrzeugmodells für Wärmemanagementbetrachtungen [nach Lund u. a. 2008, Präsentation].

grad des Gesamtsystems somit variierbar ist. Ein einfaches Innenraummodell, das als thermische Last bei der Auslegung der Klimaanlage dient, kann z.B. durch ein 3-D-CFD-Modell ersetzt werden, ohne an den übrigen Komponenten wie Kühlkreislauf oder Motor etwas ändern zu müssen. Diese Eigenschaft der hybriden Simulation ist besonders zu Beginn der Produktentwicklung nützlich, da zu diesem Zeitpunkt nicht ausreichend genaue CAD- und Messdaten zur Verfügung stehen, um mit 3-D-Verfahren verlässliche Prognosen treffen zu können. Es ist sinnvoll, zunächst mit 0-D- oder 1-D-Verfahren zu beginnen [Lund u. a. 2005] und diese zu ersetzen, wenn die notwendigen Daten bereitstehen. An den anderen Komponenten der Kosimulation müssen dabei keine Änderungen vorgenommen werden. Neben dem Vorteil, im Sinne des Front-Loadings früher Simulationen durchführen zu können, können auf diese Weise aufgrund der deutlich geringeren Rechenzeit Konzepte schneller verglichen werden [Lang u. a. 2006]. Beim Erstellen einer Kosimulation ist zu beachten, dass die Vorhersagegenauigkeit eines Modells von der Summe aus Modellierungsungenauigkeit, der Parameterungenauigkeit und numerischer Fehler abhängt [Trčka 2008]. Die ersten beiden Fehlerquellen sind in Abbildung 1.4 als Funktion der Modellkomplexität dargestellt. Während die Modellierungsungenauigkeit mit steigender Komplexität sinkt, steigt die Parameterungenauigkeit an, da die Anzahl von Parametern, für die präzise Werte vorgegeben werden müssen, zunimmt. Der Grad der Parameterungenauigkeit ist dabei abhängig von der Kenntnis des Systems. Werden die Abweichungen addiert, zeigt sich, dass es einen Komplexitätsgrad gibt, bei dem die Gesamtabweichung ein Minimum aufweist; es ist also nicht sinnvoll, die Systemmodelle über diesen Punkt hinaus zu verfeinern.

## 1.1. Stand der Wissenschaft

Simulationsmodelle für komplexe Systeme können nach Tegethoff u.a. [2006a] entweder als geschlossenes Modell mit einer geeigneten Modellierungssprache oder mit gekoppelten Ein-

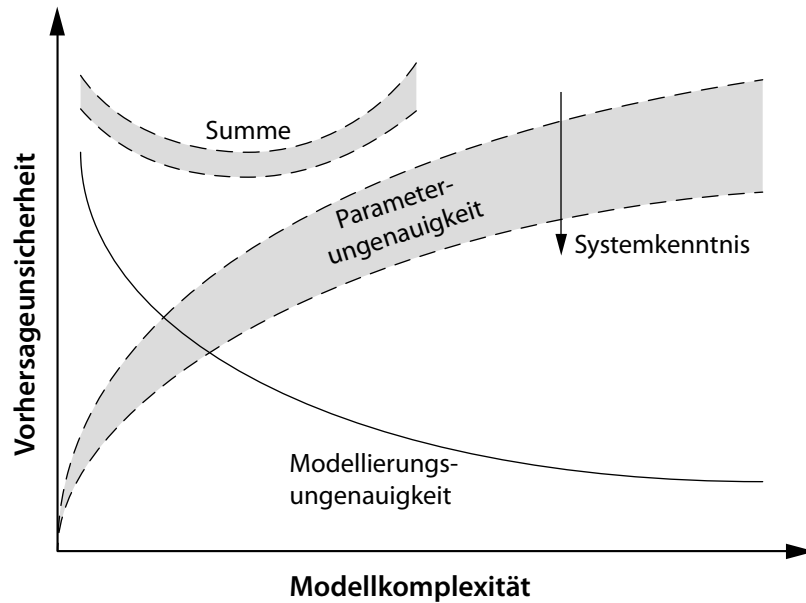


Abbildung 1.4.: Vorhersageunsicherheit von Simulationen in Abhängigkeit von der Modellkomplexität [nach Trčka 2008].

zelmodellen im Rahmen einer hybriden Simulation umgesetzt werden.

In den letzten Jahren wurden spezielle Modellierungssprachen entwickelt und bestehende erweitert, damit Gesamtsysteme modelliert werden können, deren Komponenten unterschiedlichen technischen Disziplinen zuzuordnen sind. Beispiele für solche Sprachen sind Modelica [Tiller 2001, Fritzson 2004] und VHDL-AMS [IEEE 1076.1 1999]. Für die Modellerstellung und Simulation von PKW-Klimaanlagen haben sich die deutschen Fahrzeughersteller auf ein Werkzeug geeinigt, das auf der Sprache Modelica basiert. Dieser Schritt wurde als notwendig erachtet, da die verschiedenen vorher eingesetzten Programme die Austauschbarkeit und Verschaltung unterschiedlicher Komponentenmodelle stark behinderten [Schneider u. a. 2005]. Kritiker dieses Vorgehens verweisen darauf, dass auch wenn es mit den fachübergreifenden Sprachen und Programmen möglich sei, geschlossene Modelle des zu untersuchenden Gesamtsystems – z.B. eines Fahrzeugs – zu erstellen, eine Umstellung der Modelle sämtlicher Fachabteilungen auf eine solche Lösung wegen des hohen Aufwands bereits aus wirtschaftlichen Gründen nicht in Frage käme [Fujimoto 1999]. Besonders der Erhalt des Expertenwissens für die Modelle und die eingesetzten Simulationsprogramme spreche für eine Weiterverwendung der bestehenden Modelle [Kübler u. Schiehlen 2000b]. Durch die Neuerstellung und Validierung sämtlicher Modelle würden zudem erhebliche Kosten anfallen, die die Kosten der Softwareanschaffung normalerweise deutlich überschritten [Klump 2008].

Für den Einsatz hybrider Simulation sind in der Literatur zwei Ansätze zu finden: (1) Die Verbindung von wenigen (meistens zwei) Simulationsprogrammen über Spezialverbindungen und (2) allgemeine Ansätze, die auch die Kopplung beliebig vieler Teilsimulationen erlauben.

Am weitesten verbreitet ist die hybride Simulation mit zwei Programmen (Ansatz 1). Die hierfür verwendeten Spezialverbindungen sind entweder kommerziell erhältlich oder wurden

gesondert entwickelt und genau auf den Einsatzzweck abgestimmt. In der Literatur häufig zu findende Anwendungsgebiete sind z.B. elektronische integrierte Schaltungen [z.B. Newton u. Sangiovanni-Vincentelli 1983, Gelinas u. a. 1996, Fummy u. a. 2007], Produktionsketten [z.B. Straßburger u.a. 2003, Raab u. a. 2008], Gebäudesimulation [z.B. Yahiaoui u. a. 2003, Perschk u. Meinholf 2007], Verbindung von FEM-/CFD-Werkzeugen mit 0-D-/1-D-Werkzeugen [z.B. Wünsche u.a. 1997a, Tonchev u. a. 2007], Anbindung eines in Simulink modellierten Reglers an ein Systemmodell [z.B. Arnold 2004, Nötges u. a. 2007] und die Einbindung echter Hardware in Simulationen [z.B. Le Marrec u. a. 1998, Köhler u. a. 2008].

Seit den 1980er Jahren werden Methoden und Software für den allgemeinen Ansatz (2) entwickelt, um mehrere Simulationsprogramme über ein Netzwerk verteilt zu koppeln. Hierbei ist das US-amerikanische Verteidigungsministerium als treibende Kraft hervorzuheben: 1983 wurde das SIMNET-Projekt (SIMulator NETworking) ins Leben gerufen [Miller u. Thorpe 1995], aus dem die Kosimulationsstandards DIS [IEEE 1278.1 1996], ALSP [Weatherly u. a. 1993] und schließlich HLA [IEEE 1516.3 2003] hervorgingen. Auch wenn etwa Straßburger u.a. [2001] HLA als die zurzeit fortschrittlichste Kopplungstechnologie bewerten, wird sie in der Industrie – mit wenigen Ausnahmen – nicht eingesetzt [Boehr 2005]. Nach Taylor u. a. [2002] liegt dies daran, dass HLA mit Blick auf die Anforderungen des Militärs entwickelt wurde, die Industrie aber andere und im Allgemeinen deutlich geringere Anforderungen stellt. Insbesondere sind Kopplungsprojekte in der Industrie vom Umfang deutlich kleiner, so dass der Einsatz der komplexen HLA-Umgebung wirtschaftlich nicht sinnvoll ist [Trčka 2008]. Andere Umgebungen beschäftigen sich mit eingebetteten Systemen (z.B. Ptolomy II [Lee 2003]) oder konzentrieren sich auf einzelne technische Disziplinen (z.B. Exite Ace [Extessy 2010]). Disziplinübergreifende Kosimulationen, die auch detaillierte Modelle thermischer Teilsysteme einschließen, sind in der Literatur nur wenig vertreten. Nach bestem Wissen des Autors sind derzeit lediglich zu zwei Kosimulationsplattformen Veröffentlichungen zu finden: ICOS [Puntigam u.a. 2006b] und TISC [Kossel u.a. 2009a] – wobei TISC der Name der im Rahmen dieser Arbeit entwickelten Kosimulationsumgebung ist, die auch in der Industrie bereits zum Einsatz kommt [siehe z.B. Maué u. Kauf 2006, Lund u. a. 2008]. Im Rahmen diverser Dissertationen wurden bereits Kosimulationsumgebungen erstellt und untersucht. Sie können als Untergruppe des allgemeinen Kosimulationsansatzes gesehen werden, da die gewählten Ansätze zwar allgemeingültig sein sollen, die entwickelten Lösungen aber oft mit Blick auf zwei oder drei Simulationsprogramme erstellt wurden, um spezielle Effekte aus dem Arbeitsbereich des jeweiligen Instituts zu untersuchen. Beispiele hierfür sind Kübler [2000], Boehr [2005], Dreier [2006] und Trčka [2008]. Die in diesen Arbeiten erzielten Ergebnisse sind prinzipiell zwar auf weitere Simulationsprogramme erweiter- und übertragbar, es werden jedoch spezielle Eigenschaften der bereits angebundenen Programme verwendet, so dass die Erweiterbarkeit und Übertragbarkeit immer überprüft werden muss.

Ein Prozess, wie in der Industrie eine Kosimulation aufzubauen und durchzuführen ist und welche Aufgaben den verschiedenen Personen zufallen, ist nach Kenntnis des Autors bisher nicht veröffentlicht.



Zur Simulation der Teilsysteme von Linien- oder Reisebussen sind wenige Veröffentlichungen zu finden. Beispiele sind Köhler u. a. [1990], in der ein Innenraummodell beschrieben wird, und Ackerl u. a. [2010], in der der Fokus auf dem Antriebsstrang liegt. Simulationen des Gesamtsystems Reisebus mit detaillierten Modellen der verschiedenen Teilsysteme, insbesondere des thermischen Systems, sind dem Autor nicht bekannt.

## 1.2. Ziele der Arbeit

Im Rahmen dieser Arbeit werden die Folgenden, im Stand der Wissenschaft aufgezeigten Lücken geschlossen:

- ♦ **Umgebung für die hybride Simulation thermischer Systeme**

Im Rahmen dieser Arbeit wird eine Umgebung für hybride Simulation entwickelt, mit deren Hilfe Simulationen verschiedener Modelle zu einer Gesamtsystemssimulation verbunden werden können.

Um die Lücken in Bezug auf eine nachhaltige allgemeine Kosimulation und den Arbeitsprozess zu schließen, wurde in dieser Arbeit die Blickrichtung von der Industrie aus gewählt. Durch Gespräche mit Entwicklungs- und Forschungsabteilungen verschiedener Firmen wurde festgestellt, welche Simulationsprogramme im Einsatz sind. Die Auswahl der zu unterstützenden Programme konzentriert sich auf das Gebiet der thermischen Simulation und schließt die in der Industrie verwendeten Programme ein. Da es sich hierbei um kommerzielle Programme handelt, können deren Schnittstellen bei der Anbindung an die Kosimulation nicht angepasst oder erweitert werden [Nicol u. Heidelberger 1996]. Dies wirkt sich auf den möglichen Funktionsumfang aus und muss bei der Entwicklung der Kosimulationsumgebung berücksichtigt werden. So muss z.B. davon ausgegangen werden, dass einzelne Zeitschritte nicht wiederholt werden können und keinerlei Zugriff auf modellinterne Zustandsvariablen oder die Arbeitsweise des Löser möglich ist [vgl. Kübler u. Schiehlen 2000a]. Weil auch Simulatoren verbunden werden sollen, die auf unterschiedlichen Betriebssystemen ausgeführt werden, muss die Kosimulationsumgebung plattformunabhängig ausgeführt sein.

Durch die hybride Simulation auftretende Effekte und deren Auswirkungen auf Simulationsgenauigkeit und -dauer werden an Hand einfacher Beispiele aufgezeigt. Es werden neue Verfahren entwickelt, mit deren Hilfe die negativen Auswirkungen minimiert oder vermieden werden können. Anders als bei der verteilten Simulation wird die Kosimulation in dieser Arbeit nicht vorrangig dazu eingesetzt, die Modellberechnung zu beschleunigen, sondern dazu, die Simulation komplexer Gesamtsysteme überhaupt zu ermöglichen. Eine Bewertung des erzielten Geschwindigkeitsgewinns oder -verlustes steht somit nicht im Vordergrund und ist zudem praktisch nicht möglich, da eine solche Bewertung immer den Aufbau eines geschlossenen Vergleichssystems bedingt; dies ist jedoch bei Verwendung unterschiedlicher Programme für die Modellierung nicht möglich.

### ♦ **Arbeitsprozess zum Aufbauen und Durchführen einer Gesamtsimulation**

Im Rahmen dieser Arbeit wird ein Arbeitsprozess entwickelt, in dem die Modellentwickler die Modelle für die Kosimulation liefern, die Simulation jedoch nicht durchführen. Dies entspricht dem Vorgehen, das in der industriellen Praxis häufig anzutreffen ist. Der Anwender der Kosimulation verfügt häufig über keine oder nur rudimentäre Kenntnisse über die Wirkweise der eingesetzten abteilungsspezifischen Simulationsprogramme, weshalb die Modelle in dieser Arbeit als unbekannt und unveränderbar angesehen werden – lediglich das prinzipielle physikalische Verhalten wird als bekannt vorausgesetzt. Eingriffe in das Verhalten der Modelle während einer Simulation sind somit nur über die Kosimulationsschnittstelle möglich. Dem Modellersteller fällt damit die Aufgabe zu, das Modell für die Kosimulation vorzubereiten; ein Kosimulationsverantwortlicher stellt dieses anschließend den einzelnen Fachabteilungen bereit.

### ♦ **Gesamtsimulation des thermischen Systems eines Reisebusses**

Als Anwendung wird die Simulation des thermischen Systems eines Reisebusses mit hybrider Simulation vorgestellt. Hierzu werden detaillierte Modelle der Teilsysteme Kältekreislauf (mit dem Kältemittel R-744), Kühl-/Heizkreislauf, Innenraum und Längsdynamik erstellt und mit zwei zusätzlichen Modellen zur Bestimmung des Fahr- bzw. Umgebungszustands im Rahmen einer Kosimulation berechnet. Nach einer Untersuchung der dynamischen Eigenschaften des gekoppelten Systems wird eine virtuelle Fahrt betrachtet, die die realen Einsatzbedingungen besser nachbildet [Daimler AG 2007] als die üblicher Weise untersuchten stationären Randbedingungen, die kritische Betriebszustände erwarten lassen [vgl. Lund u. a. 2005]. Während der virtuellen Fahrt werden unterschiedliche Strategien zur Regelung der Innenraumtemperatur in Hinblick auf mögliche Energieeinsparungen betrachtet und verglichen.

## 1.3. Aufbau der Arbeit

Kapitel 2 gibt einen Überblick über die wichtigsten Aspekte der Softwarekopplung: die Kommunikation und die Synchronisation.

Auf ausgewählte numerische Effekte, die während einer Kosimulation auftreten können, geht Kapitel 3 ein. Neben deren Beschreibung werden dabei auch Wege vorgestellt, wie die Effekte abzumildern oder gänzlich zu vermeiden sind.

Die Struktur und die Umsetzung der im Rahmen dieser Arbeit erstellten Kosimulationsumgebung wird in Kapitel 4 vorgestellt.

Einen Arbeitsprozess, der den Weg von der Modellerstellung bis zur Durchführung der Kosimulation beschreibt, präsentiert Kapitel 5. Er ist als Ausgangspunkt für komplexere Prozesse zu verstehen.

Dem Arbeitsprozess aus Kapitel 5 folgend wird in Kapitel 6 die Kosimulation des thermischen Gesamtsystems eines Reisebusses vorgestellt. Hierfür werden zunächst die Teilsysteme identifiziert und modelliert. Die Teilmodelle werden anschließend durch Kopplung zu einem Gesamtmodell zusammengeschlossen und in einer Kosimulation berechnet. Das Hauptaugenmerk liegt hierbei weniger auf den konkreten Ergebnissen, sondern auf der Methodik, die Kosimulation aufzubauen, um den Reisebus in der Form überhaupt abbilden zu können. Aus diesem Grund wird ein Vorgehen gewählt, dass die Komplexität des Gesamtmodells erhöht, jedoch nah an der industriellen Praxis liegt.

Die Zusammenfassung dieser Arbeit ist in Kapitel 7 zu finden.



## 2. Ausgewählte Aspekte der Kopplung

Der Begriff „*Kopplung*“ wird in den unterschiedlichen Bereichen der Computersimulation verschieden verwendet und beschreibt das Vorgehen, Programme oder Programmteile zu verbinden; über die Art der Programme, die eingesetzte Technik sowie Umfang und Methode von Kommunikation und Interaktion wird hierbei keine Aussage getroffen. Eindeutige Bezeichnungen für die einzelnen Kopplungsmethoden konnten sich in der Literatur bisher nicht durchsetzen [Geimer u. a. 2006].

Dieses Kapitel klärt die Terminologie für diese Arbeit und stellt wesentliche Aspekte und Techniken der Kopplung vor. Die konkrete Implementierung der entwickelten Kosimulationsumgebung wird in Kapitel 4 beschrieben.

### 2.1. Klassifizierung von Modellbildung und Kopplung

Bei der Erstellung der Modelle für ein disziplinübergreifendes System muss zunächst geklärt werden, ob die Modellbildung und die anschließende Simulation jeweils geschlossen oder verteilt erfolgen soll. Abhängig hiervon können verschiedene Kopplungsmethoden verwendet werden. Die folgenden zwei Abschnitte sind untergliedert in geschlossene und verteilte Modellbildung und gehen auf die jeweils möglichen Kopplungsmethoden ein. Abbildung 2.1 stellt die in diesem Kapitel vorgestellte Klassifizierung anschließend grafisch dar.

#### **Geschlossene Modellbildung**

Bei der geschlossenen Modellbildung wird ein einzelnes Simulationsprogramm verwendet, um ein Modell für das zu untersuchende System zu erstellen. Die meisten Simulationsprogramme entstammen einer einzelnen technischen Disziplin und sind Mittel der Wahl, wenn Systeme der jeweiligen Disziplin modelliert werden. Disziplinübergreifende Systeme können mit diesen Programmen oft nicht oder nur rudimentär abgebildet werden. Aus diesem Grund wurden Erweiterungen zu bestehenden Programmen und Sprachen wie VHDL-AMS [IEEE 1076.1 1999] und Simscape [Simscape 2010] sowie spezielle Sprachen wie Modelica [Tiller 2001, Fritzson 2004] entwickelt. Hierbei ist allerdings zu beachten, dass auch diese Lösungen und Erweiterungen vorwiegend auf dem Gebiet eingesetzt werden können, aus dem sie entstammen [Hessel u. Haase 2009] – im Fall von VHDL-AMS z.B. der Elektrik –, auch wenn die Zahl der

verfügbaren Bibliotheken für die Sprachen stetig steigt. Für die Sprache Modelica existiert bereits eine Vielzahl freier und kommerzieller Komponenten-Bibliotheken, die verschiedenen technischen Disziplinen entstammen (eine Übersicht gibt Modelica [2009]). Die Entwicklung weiterer Modelica-Bibliotheken wird auch im Rahmen geförderter Projekte vorangetrieben. Ein Beispiel ist das von der Europäischen Union geförderte EUROSYSLIB-Projekt, in dessen Rahmen freie und kommerzielle Bibliotheken entwickelt werden [EUROSYSLIB 2010].

Das mit einem der beschriebenen Werkzeuge erstellte Systemmodell kann entweder geschlossen oder verteilt simuliert werden. Durch die Aufteilung des Gesamtmodells in Teilmodelle können diese auf weitere Ressourcen (Prozessoren, Arbeitsspeicher, ...) verteilt werden, wodurch die für die Simulation benötigte Zeit potentiell gesenkt werden kann. Sind die zusätzlichen Ressourcen Teil des selben Computers, wird dies „paralleles Rechnen“ genannt, sind sie über ein Netzwerk verteilt „verteiltes Rechnen“.

Typische Beispiele für eine geschlossene Modellbildung mit verteilter Simulation sind die Finite-Elemente-Methode (FEM) und die numerische Fluidodynamik (CFD), bei denen große monolithische Modelle erstellt werden, die gut parallelisierbar sind.

### **Verteilte Modellbildung**

In der industriellen Praxis werden die Modelle für die einzelnen Teilsysteme häufig von unterschiedlichen Arbeitsgruppen erstellt. Besonders in großen Unternehmen entwickeln die verschiedenen Fachabteilungen Modelle der entsprechenden Teilsysteme. Hierfür kommen Simulationsprogramme zum Einsatz, die auf das jeweilige Arbeitsgebiet spezialisiert sind. Da mit diesen Programmen ein disziplinübergreifendes System nicht oder nur sehr rudimentär abgebildet werden kann, bietet sich für die Modellierung des Gesamtsystems eine Kopplung verschiedener Spezialprogramme an. Hierdurch können die speziellen Eigenschaften der Programme genutzt werden, um detaillierte Modelle zu erstellen und effizient zu berechnen, während gleichzeitig die Wechselwirkungen mit anderen Teilsystemen berücksichtigt werden können. Dass bereits bestehende Modelle weitergenutzt werden können, ist ein wirtschaftlich nicht zu vernachlässigender Vorteil.

Ob das Gesamtsystemmodell geschlossen oder verteilt berechnet werden kann, hängt von der verwendeten Art der Kopplung ab. Je nach Integrationsart der Teilsysteme können Kopplungen unterteilt werden in Modell-, Solver- und Prozess-/Programmverbund [Gärtner u. a. 2000, Busch 2006]. Die Eigenschaften der einzelnen Kopplungsarten und Beispiele für diese sind:

#### **♦ Modellverbund**

In einem Modellverbund werden Modelle in einem Simulationsprogramm zusammengeschlossen, die zuvor aus den entsprechenden Simulatoren exportiert wurden. Das Programm, in das die Modelle eingebunden werden, wird allgemein als „Master-Simulator“ bezeichnet. Durch das Lösen des Gesamtsystems durch den Master-Simulator kann es

passieren, dass für das eingebundene Fremdmodell eine andere Lösung berechnet wird oder es überhaupt nicht gelöst werden kann [Arnold 2004].

Beispiele für einen Modellverbund finden sich in Nötges u. a. [2007] und Richert u. a. [2005]. Hierbei werden mit der Sprache Modelica erstellte Teilmodelle über eine kommerzielle Schnittstelle in das Programm Simulink eingebunden. Riederer u. a. [2009] stellen eine Methodik vor, mit der Trnsys-Komponenten in Simulink verwendet werden können und umgekehrt.

#### ♦ **Solververbund**

Werden bei einem Modellverbund für die einzelnen Teilsysteme eigenständige Gleichungslöser verwendet, wird hierdurch ein Solververbund gebildet. Der Löser des Master-Simulators bestimmt bei diesem Verfahren lediglich die Schrittweite, mit der die verschiedenen Löser kommunizieren. Der Solververbund stellt die am weitesten verbreitete Verbundart dar, meistens verwendet, um zwei in unterschiedlichen Programmen erstellte Modelle zusammenzuschließen.

Ein Beispiel für einen Solververbund wird in Lund u. a. [2005] präsentiert. Hierbei wird ein in VBA programmiertes Skript verwendet, um die allgemeinen Schnittstellen der Simulationsprogramme Kuli und Flowmaster anzusprechen. Das Skript avanciert hierdurch zum Master-Simulator. Regner u. a. [2002] rufen zu jedem Zeitschritt einer Simulation mit CRUISE das Programm BOOST auf, um aktualisierte Motorkennfelder zu generieren und mit diesen in CRUISE weiterrechnen zu können. Schröder u. a. [2002] binden ein stationär rechnendes 1-D-Simulationsprogramm in eine transiente 3-D-Simulation ein. Alle drei Veröffentlichungen nutzen Spezialschnittstellen für das Ansprechen der einzelnen Simulationsprogramme.

#### ♦ **Programm-/Prozessverbund („Kosimulation“)**

Beim Programm- oder Prozessverbund werden die verschiedenen Teilmodelle in den entsprechenden Simulatoren berechnet. Die Simulatoren werden in getrennten Prozessen gestartet und die einzelnen Löser durch Kommunikation während der Simulation verbunden. Dass bei der Kopplung der verschiedenen Teilmodelle mehr Ressourcen genutzt werden können und damit ggf. ein Geschwindigkeitsvorteil erzielt werden kann, ist nicht Ziel, aber ein positiver Nebeneffekt. Der Name „*Kosimulation*“ wird als Synonym für den Programm-/Prozessverbund verwendet.

Als Veröffentlichungen zur Funktionsweise von Kosimulationen sind vor allem Dissertationen zu finden: Kübler [2000] erweitert das Simulationssystem NEWMOS, um die Kopplung verschiedener Programme zu ermöglichen, und setzt Schnittstellen zu SIMPACK und Simulink um. In Boehr [2005] wird eine Kosimulationsumgebung namens FAMAS erstellt, um Logistikprozesse abzubilden. Dreier [2006] erstellt die Umgebung INTERACT, die eine Kopplung von Simulink mit auf eingebetteter Elektronik ausgeführten Simulationen erlaubt. Trčka [2008] entwickelt eine Kosimulation zur Berechnung von Gebäuden mit unterschiedlichen Lüftungssystemen, wobei die Kopplung auf einen Computer beschränkt ist.

Veröffentlichungen aus der Industrie sind für den thermischen Bereich rar gesät wie schon der Titel von Lund u. a. [2008] – „*Innovation durch Co-Simulation!*“ – vermuten lässt. In dieser Veröffentlichung wird die Notwendigkeit der Kosimulation für die Berechnung komplexer Systeme wie eines Gesamtfahrzeugs dargelegt und der Einsatz von TISC für die Simulation eines aus mehr als 18 Teilmodellen bestehenden Gesamtmodells beschrieben. Beiträge zur Kosimulationsumgebung ICOS [z.B. Puntigam u.a. 2006b und Lang u. a. 2006] stellen eine Gruppe von Veröffentlichungen dar, die sich ebenfalls mit der Berechnung eines Gesamtfahrzeugmodells beschäftigt.

Die Kosimulation – der Programm-/Prozessverbund – ist die Kopplungsart, die im Rahmen dieser Arbeit betrachtet wird.

### **Grafische Darstellung der Klassifizierung von Kopplungsverfahren**

Abbildung 2.1 zeigt die in diesem Kapitel vorgestellte Klassifizierung. Hierbei setzt sich das dargestellte disziplinübergreifende System beispielhaft aus den Bereichen Mechanik, Thermodynamik und Elektronik zusammen. Nach der Aufteilung in die Modellbildung mit einem oder mehreren verschiedenen Simulationsprogrammen (geschlossene bzw. verteilte Modellbildung) wird zwischen geschlossener und verteilter Simulation unterschieden. Die unterste Ebene in der Abbildung stellt die unterschiedlichen Kopplungsmethoden dar. Da sich die weiteren Ausführungen in dieser Arbeit ausschließlich auf den Programm-/Prozessverbund beziehen, ist dieser hervorgehoben dargestellt.

Eine andere Art der Gliederung von Kopplungsmethoden als die hier vorgestellte ist die Unterteilung in starke und schwache Kopplung. Hiermit wird beschrieben, wie eng die Verbindung der einzelnen Teilmodelle während der Simulation ist. Bei der starken Kopplung werden alle Teilmodelle zu jedem Zeitschritt so lange iteriert, bis die Lösung gefunden wurde. Da dies beim Einsatz mehrerer unterschiedlicher Gleichungslöser nicht möglich ist, wird hierbei von schwacher oder loser Kopplung gesprochen. Nach dieser Einordnung stellen der Modellverbund eine starke Kopplung, Solver- sowie Programm-/Prozessverbund schwache Kopplungen dar. Die verteilte Simulation bei geschlossener Modellbildung ist immer eine starke Kopplung, da hier ein zentraler Löser zum Einsatz kommt. Trčka u. a. [2007] vergleichen die Genauigkeit und die benötigte Simulationszeit für unterschiedliche Kopplungen des selben Systems. Hiernach lässt die schwache Kopplung die höchste Simulationsgeschwindigkeit bei bester Genauigkeit erhoffen.

## **2.2. Synchronisation**

Die Aufgabe der Synchronisation ist das Festlegen der Zeitpunkte in einer Kosimulation, zu denen die einzelnen Programme Informationen austauschen. Hierdurch wird auch sichergestellt, dass die Simulationszeiten aller beteiligten Programme abgeglichen werden. In den



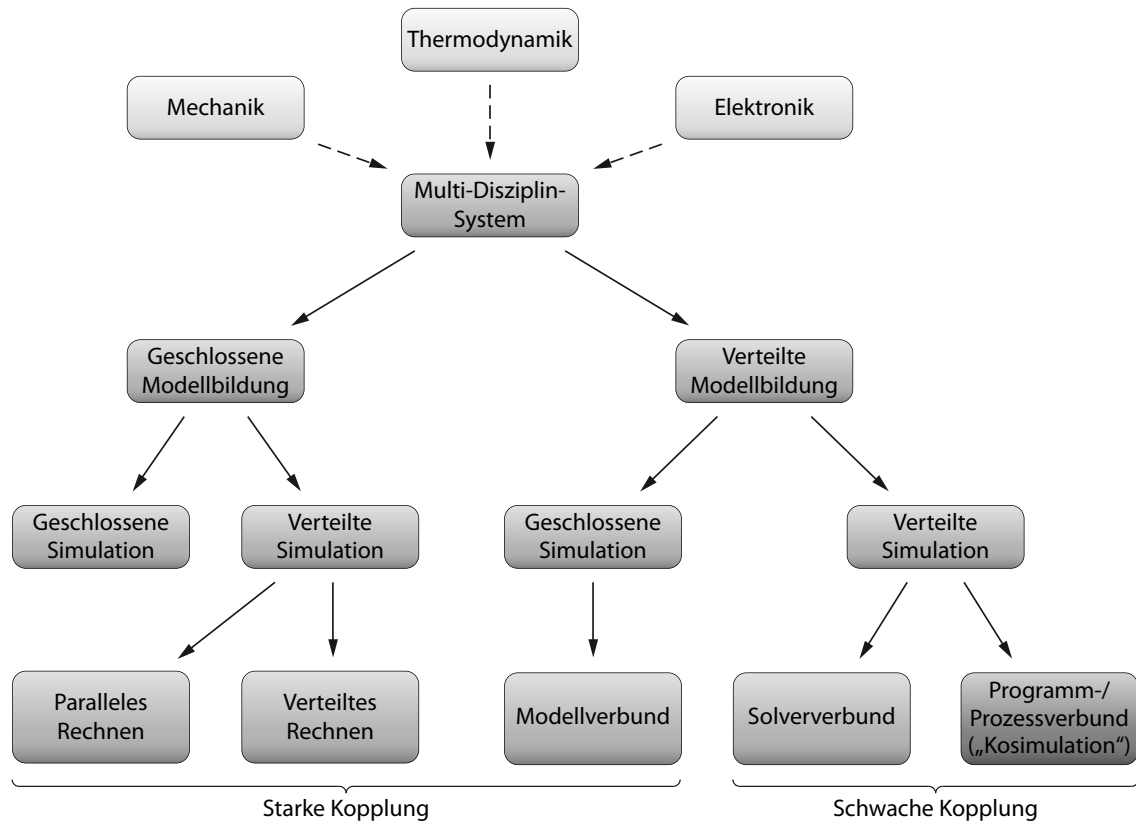


Abbildung 2.1.: Klassifizierung der Modellbildung und Kopplung disziplinübergreifender Systeme

1970er Jahren entwickelten Bryant [1977] und Chandy u. Misra [1979] mit die ersten Algorithmen, die sich auf die Synchronisation von parallelen Simulationen mit diskreten Ereignissen (PDES – Parallel Discrete Event Simulation) beziehen [Fujimoto 1990]. Nach der Literatur werden Synchronisationsmethoden in die folgenden Kategorien eingeordnet:

#### ♦ Konservative Verfahren

Die Simulationszeit schreitet in allen beteiligten (Teil-)Simulationen stetig und mit gleicher Zeitschrittweite voran. Bei jedem auftretenden Ereignis muss der entsprechende logische Prozess warten, bis sichergestellt ist, dass vor diesem Ereignis keine neuen Daten empfangen werden müssen. Treten während des gleichen Zeitschritts in mehreren Teilsystemen Ereignisse auf, kann dies Blockierungen verursachen, da mehrere Prozesse aufeinander warten. Das konservative Verfahren wird in Chandy u. Misra [1979] vorgestellt. Hier werden auch erste Ansätze zur Vermeidung möglicher Blockierungen durch gegenseitige Abhängigkeiten dargelegt.

Eine Untergruppe stellt das von DeBenedictis u. a. [1991] vorgestellte konservative, asynchrone Verfahren dar. Während alle Teilsysteme hierbei wie bei klassischen konservativen Verfahren zu festgelegten Zeitintervallen Informationen austauschen, dürfen die einzelnen Prozesse mit einer individuellen Zeitschrittweite für die Integration arbeiten.

### ♦ Optimistische Verfahren

Hierbei wird davon ausgegangen, dass zu jedem Zeitpunkt sämtliche Informationen vorliegen und synchronisiert werden darf. Stellt sich später heraus, dass diese Annahme nicht korrekt war, z.B. weil in einem anderen Teilsystem ein Ereignis auftritt, wird die Simulationszeit in allen (Teil-)Simulationen bis zum letzten Zeitpunkt zurückgesetzt, für den konsistente Werte vorliegen, und von dort neu gerechnet.

Übersichten über verschiedene Verfahren und tiefergehende Betrachtungen sind in der Literatur zu finden [z.B. in Ferscha 1996, Ghosh 2002, Fujimoto 2003, Dreier 2006, Puntigam u.a. 2006a und Trčka 2008].

Während sich optimistische Verfahren in der parallelen und verteilten Simulation durchgesetzt haben [Fujimoto u. a. 2007], stellen sie bei der Kosimulation eine große Hürde bei der Anbindung von Simulationsprogrammen dar. Wesentliche Voraussetzung für die Verwendung optimistischer Verfahren ist die Möglichkeit, die Simulationszeit zurücksetzen zu können. Diese so genannten Roll-Back-Verfahren sind mindestens im thermischen Bereich in den wenigsten Simulationsprogrammen implementiert. Eine nachträgliche Implementierung erfordert erheblichen Aufwand [Page u. a. 1999], der zudem nicht vom Entwickler der Kosimulationsumgebung, sondern dem des Simulationsprogramms zu leisten ist. Da dieser durch die Implementierung jedoch keinen direkt ersichtlichen Vorteil hat, ist die Motivation für die Umsetzung generell als sehr gering einzustufen. Zudem ist für den Einsatz optimistischer Verfahren ein Zugriff auf die internen Zustände des Systems und die Ereignisgenerierung notwendig, was bei den meisten Simulationsprogrammen nicht gegeben ist [Dreier 2006, Puntigam u.a. 2006b]. Für allgemeine Kosimulationen stellen somit konservative Verfahren die einzig mögliche Wahl dar. Da die einzelnen Gleichungslöser im Allgemeinen mit variabler Zeitschrittweite arbeiten, wird das Verfahren immer ein asynchrones sein.

Im Folgenden werden zunächst die parallele und die serielle Synchronisation vorgestellt. Anschließend wird die Wahl der Synchronisationsschrittweite diskutiert.

### 2.2.1. Parallele Synchronisation

Bei der parallelen Synchronisation berechnen alle Simulationsprogramme gleichzeitig den aktuellen Zeitschritt  $i$ . Zu dem vorher festgelegten Endzeitpunkt  $T_i$  werden sämtliche Programme angehalten und die Daten ausgetauscht. Anschließend wird der nächste Zeitschritt berechnet. Abbildung 2.2 zeigt dieses Synchronisationsschema.

Durch die Ereignisgenerierung für die einzelnen Synchronisationszeitpunkte wird der Berechnungsaufwand erheblich erhöht. Auf Grund der Parallelläufigkeit der beteiligten Simulationsprozesse kann dies ggf. kompensiert werden, bei Systemen mit stark unterschiedlichen Zeitkonstanten ist auch eine Verkürzung der Gesamtsimulationszeit möglich. Ereignisse, die während eines Zeitschritts auftreten, können erst am Ende an die anderen Clients propagiert werden.

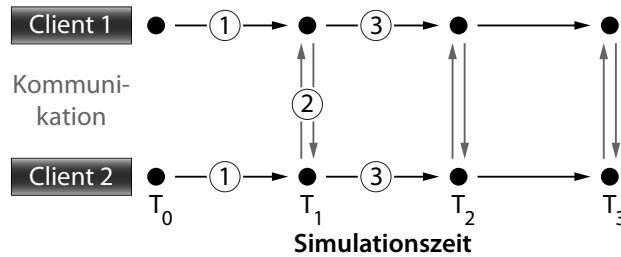


Abbildung 2.2.: Parallele Synchronisation.

In der Literatur werden für ähnliche Synchronisationskonzepte auch die Begriffe *Zwiebel* [Hensen 1999] und *voll-dynamisch* [Zhai 2003] verwendet.

### 2.2.2. Serielle Synchronisation

In Verfahren mit serieller Synchronisation rechnen die Clients nacheinander bis zu dem vorher festgelegten Zeitpunkt  $T_i$ . In Abbildung 2.3 simuliert beispielsweise *Client 1* zuerst von  $T_0$  bis  $T_1$  (1), sendet die Ergebnisse an *Client 2* (2), der daraufhin ebenfalls von  $T_0$  bis  $T_1$  rechnet (3). Abschließend werden die Daten von *Client 2* an *Client 1* gesendet (4), der mit dem nächsten Zeitschritt fortfährt (5).

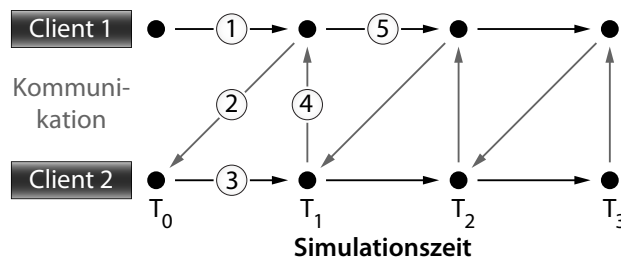


Abbildung 2.3.: Serielle Synchronisation.

Mit diesem Verfahren wird es den Clients ermöglicht, auf Ereignisse zu reagieren, die in vorher aufgerufenen Clients aufgetreten sind. Wird z.B. in Client 1 während der Berechnung des aktuellen Zeitschritts ein Schaltvorgang vorgenommen, kann die Information über dessen genauen Zeitpunkt zusätzlich an Client 2 übertragen werden, der hierdurch angemessen auf den Vorgang reagieren kann. In die andere Richtung ist dies nicht möglich, da hierfür ein Zurücksetzen der Simulationszeit in Client 1 notwendig wäre. Für kontinuierliche Signale können geeignete Interpolationsverfahren verwendet werden, um den transienten Verlauf nachzubilden, da neben den aktuellen Werten in Client 2 auch die zukünftigen Werte von Client 1 bekannt sind. Die Rückwirkung erfolgt jedoch wie bei der parallelen Synchronisation (Kapitel 2.2.1) zeitversetzt.

Die Reihenfolge, in der die Simulationsprogramme aufgerufen werden, sollte anhand der physikalischen Informationsflüsse bestimmt werden [Puntigam u.a. 2006a]. Gerade bei Simulationen

mit Teilsystemen, die mit mehreren anderen Systemen in Wechselwirkung stehen, kann jedoch kein eindeutiger Informationsfluss abgeleitet werden. Zudem müssen bei jeder Änderung der Variablen, die von jedem Modell gesendet und empfangen werden, und bei jedem Hinzufügen, Entfernen und Austauschen von Modellen die Abhängigkeiten neu bestimmt werden.

In der Literatur werden für die serielle Synchronisation auch die Begriffe *Ping-Pong* [Hensen 1999], *quasi-dynamisch* [Zhai 2003] und *sequentiell* [Puntigam u.a. 2006a] verwendet.

### 2.2.3. Synchronisationsschrittweite

Bei der Kosimulation wird neben dem Datenaustausch viel Zeit für die Synchronisation der einzelnen Simulatoren benötigt [Kim u. a. 2002]. Da zudem das häufige Reinitialisieren der Löser einen großen Leistungseinbruch zur Folge hat [Arnold 2004], gilt es, die Anzahl der Kommunikationsschritte so klein wie möglich bzw. die durchschnittliche Kommunikations-schrittweite so groß wie möglich zu wählen. Diesem Ziel entgegen wirkt eine Vergrößerung des eingebrachten Fehlers – der Abweichung von der Lösung eines geschlossenen Modells – bei Erhöhung der Schrittweite.

Nach Peacock u. a. [1979] wird das zeitliche Verhalten der einzelnen logischen Prozesse (LP) beim parallelen und verteilten Rechnen eingeteilt in:

- ♦ **Synchrone LP-Simulation**

Alle Teilsysteme verwenden dieselbe Schrittweite für die Iteration. Hierbei wird nach jedem Zeitschritt die minimale Schrittlänge für das nächste Intervall bestimmt und an alle Prozesse übermittelt. Durch dieses Vorgehen wird sichergestellt, dass alle Teilsysteme direkt auf auftretende Ereignisse (sofern diese vorhersagbar sind) reagieren können.

- ♦ **Asynchrone LP-Simulation**

Die einzelnen Teilsysteme rechnen mit unabhängigen Schrittweiten. Bei der Synchronisation der einzelnen Prozesse können wiederum konservative oder optimistische Verfahren (vgl. Kapitel 2.2) eingesetzt werden.

Eine genauere Beschreibung ist Ferscha [1996] zu entnehmen. Übertragen auf die Kosimulation bedeutet diese Einordnung:

- ♦ **Synchrone Kosimulation**

Zum vorher definierten Zeitpunkt werden alle Clients angehalten und synchronisiert. Hiernach wird festgelegt, wie groß das nächste Synchronisationsintervall ist. Wie viele interne Zeitschritte die jeweiligen Löser verwenden ist unerheblich.

- ♦ **Asynchrone Kosimulation**

Jeder Client rechnet mit einer individuellen Zeitschrittweite und synchronisiert nach jedem Zeitschritt. Zwar wird potentiell häufiger kommuniziert als bei der synchronen Kosimulation, es müssen allerdings keine zusätzlichen Ereignisse für die Synchronisation

generiert werden. Dies schlägt sich – bezogen auf den einzelnen Prozess – in einem Geschwindigkeitsvorteil nieder.

In der Literatur sind nur sehr wenige Beispiele asynchroner Kosimulationen zu finden; Puntigam u.a. [2006b] beschreiben eines davon. Hierbei dürfen immer nur die Simulationsprozesse mit der Berechnung fortfahren, deren aktuelle Simulationszeit am kleinsten ist. Da die jeweiligen Simulationsfortschritte bekannt sind, müssen die Daten nur dann ausgetauscht werden, wenn der jeweils andere Prozess als nächstes rechnen wird, wodurch der Kommunikationsaufwand gering gehalten werden kann. Abbildung 2.4 zeigt das entsprechende Synchronisationschema für den Fall einer Kopplung zweier Prozesse, von denen einer (*Client 1*) eine konstante interne Löserzeitschrittweite von  $1,0\text{ s}$  und der andere (*Client 2*) eine konstante Schrittweite von  $0,4\text{ s}$  verwendet. Zum Zeitpunkt  $0\text{ s}$  – beim Start der Kosimulation – beginnen beide Prozesse mit der Berechnung des ersten Zeitschritts, da beide den gleichen zeitlichen Fortschritt aufweisen. Bei späteren Zeitschritten können die vom Simulationspartner bekannten Daten zur Interpolation genutzt werden, da sie einem Punkt in der Zukunft zugeordnet werden können.

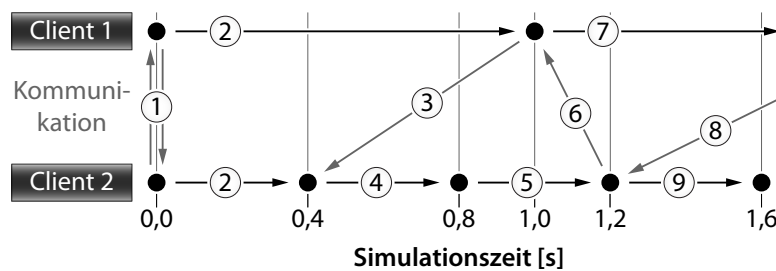


Abbildung 2.4.: Asynchrone Kosimulation nach Puntigam u.a. [2006b] (zeitlicher Ausschnitt).

Aus Abbildung 2.4 ist zu erkennen, dass es sich nach der Unterscheidung in Kapitel 2.2 prinzipiell um eine serielle Synchronisation handelt: Die einzelnen Clients rechnen nacheinander, lediglich bei identischem Simulationsfortschritt dürfen sie gleichzeitig den nächsten Zeitschritt berechnen.

Wird in dem Beispiel aus Abbildung 2.4 eine rein parallele Synchronisation nach Kapitel 2.2.1 (und somit eine synchrone Kosimulation) verwendet, ergeben sich folgende Auswirkungen (vgl. Abbildung 2.5):

#### ♦ **Parallelläufigkeit**

*Client 1* und *2* rechnen immer gleichzeitig, wodurch die für die Simulation benötigte Rechenzeit verringert wird. Je mehr Clients an der Kosimulation beteiligt sind, desto größer ist der erreichbare Geschwindigkeitsvorteil.

#### ♦ **Interpolation**

Die vom jeweils anderen Client empfangenen Werte liegen nicht mehr mit einem zukünftigen, sondern dem aktuellen Zeitstempel vor und können daher nicht für die Interpolation

genutzt werden. Je nachdem, wie stark sich die Löseschrittweiten der Clients unterscheiden, hat dies unterschiedlich große Auswirkungen. In Abbildung 2.4 kann *Client 2* von  $0,4\text{ s}$  bis  $1,0\text{ s}$  einen interpolierten Wert verwenden. *Client 1* hingegen verfügt nach Schritt 6 nur über Informationen für die nächsten  $0,2\text{ s}$  – der Wegfall der Interpolation ist für *Client 1* somit gravierender als für *Client 2*.

#### ♦ Zusätzliche Ereignisse

Bei der rein parallelen Synchronisation muss immer dann ein zusätzliches Ereignis erzeugt werden, wenn Synchronisationsrate und interne Löseschrittweite nicht identisch sind. Bei mit variabler Schrittweite arbeitenden Lösern bedeutet dies, dass für jeden Synchronisationszeitpunkt ein zusätzliches Ereignis generiert wird.

Würde die Schrittweite für die parallele Synchronisation wie in Abbildung 2.5 auf  $1,0\text{ s}$  festgelegt werden, müsste *Client 2* jede volle Sekunde ein zusätzliches Ereignis erzeugen – die interne Schrittweite von  $0,4\text{ s}$  bliebe bestehen. Alternativ könnte die Austauschrate auf einen kleineren Wert festgelegt werden. Würde z.B.  $0,4\text{ s}$  gewählt, wären in *Client 1* zusätzliche Ereignisse notwendig, in *Client 2* würde die interne Schrittweite mit der Austauschschrittweite aufeinander fallen.

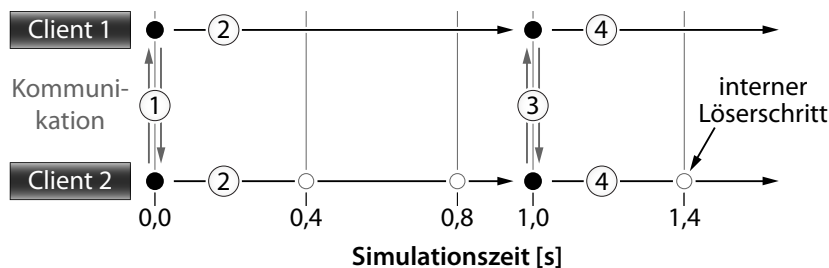


Abbildung 2.5.: Schema aus Abbildung 2.4 mit paralleler Synchronisation.

Der Absolutwert der Synchronisationsschrittweite hängt davon ab, wie schnell die Systeme auf neue Werte der gekoppelten Variablen reagieren – also den Zeitkonstanten der Systeme. Hierbei sind sowohl die Sensitivität der Empfängerseite als auch die Dynamik auf Senderseite zu berücksichtigen [Trčka 2008]. Je nach gekoppelten Modellen können die hierdurch bestimmten Schrittweiten stark unterschiedlich sein. Arnold [2004] gibt als Beispiel für ein verteiltes Mehrkörpersystem einen Wertebereich von  $0,1\text{ ms}$  bis  $10\text{ ms}$  an, Kübler [2000] einen von  $0,5\text{ ms}$  bis  $4\text{ ms}$ . In der Gebäudesimulation sind hingegen Austauschraten von 10 Minuten möglich [z.B. bei Trčka u. a. 2007].

Durch eine Änderung der Austauschrate während der Kosimulation kann die numerische Effizienz verglichen mit einer konstanten Schrittweite gesteigert werden. Kübler [2000] beschreibt eine Anpassung der Schrittweite an die aktuelle Dynamik des Systems. Eine weitere Möglichkeit ist die Beobachtung des durch die Kopplung eingebrachten Fehlers, um die Schrittweite zu kontrollieren, wie es Wünsche u.a. [1997a] vorstellen. Hierbei wird der Werteverlauf auf der Empfangsseite, der aus den zu diskreten Zeitpunkten übertragenen Werten erzeugt wird,

mit dem als korrekt angenommenen Werteverlauf auf der Sendeseite verglichen. Weichen die beiden Verläufe zu stark voneinander ab, muss der letzte Zeitschritt wiederholt werden. Dieses sog. Roll-Back-Verfahren wird jedoch von vielen Programmen nicht unterstützt. Gheorghe u. a. [2006] stellen eine Kosimulation mit serieller Synchronisation vor, bei der ein kontinuierlich rechnendes Modell mit variabler Zeitschrittweite bis zum nächsten unerwarteten Zustandsergebnis rechnet und hierauf hin die angekoppelten diskreten Simulationen bis zu diesem Zeitpunkt rechnen. Damit kein Roll-Back erforderlich ist, darf hierbei nur ein beteiligtes Modell kontinuierlich berechnet werden.

## 2.3. Kommunikation im Programm-/Prozessverbund

Während der Simulation kommunizieren die einzelnen Programme bzw. Teilmodelle miteinander, um Daten auszutauschen und den Simulationsfortschritt abzugleichen [Fujimoto 2000]. Der Umfang der Aufgaben, die die einzelnen Programme hierbei übernehmen, hängt von der gewählten Organisation der Verbindungen – der Kommunikationsstruktur – ab. Bei der Integration von Simulationsprogrammen in die Kosimulation stehen zwei Kommunikationsstrukturen zur Diskussion, die in Abbildung 2.6 schematisch dargestellt sind:

- ♦ **Direkte Verbindungen**

Kommunizieren die Programme ohne zentrale Instanz direkt miteinander, muss jede Kommunikationsschnittstelle nicht nur für den Datenaustausch, sondern insbesondere auch für die Synchronisierung des Clients mit den anderen Teilmodellen sorgen. Vorteilhaft ist, dass der Kommunikationsaufwand gering gehalten werden kann [Puntigam u.a. 2006a]. Nachteilig wirken sich die direkten Verbindungen auf den einfachen Modellaustausch aus, der in Kapitel ?? als Vorteil der Kosimulation aufgeführt ist. Wird ein Teilmodell ausgetauscht, müssen sämtliche mit dem ausgetauschten Modell kommunizierenden Modelle hierüber informiert bzw. entsprechend umkonfiguriert werden.

- ♦ **Zentraler Server**

Wird ein Server als neutrale Instanz in der Kommunikation eingesetzt, sorgt dieser nicht nur für den Austausch der Variablen zwischen den Clients, sondern übernimmt auch die Synchronisierung der einzelnen Simulationsprogramme. Hierdurch wird der Funktionsumfang und damit der Entwicklungs- und Wartungsaufwand der einzelnen Programmschnittstellen deutlich verringert. Zusätzlich muss die Konfiguration der einzelnen Teilmodelle nicht geändert werden, wenn Modelle zur Kosimulation hinzugefügt oder entfernt werden, wodurch die Konfiguration und Erweiterung der Gesamtsimulation vereinfacht wird.

Neben der Kommunikationsstruktur stellt auch die Wahl der für die Kommunikationsverbindung eingesetzten Technik eine grundlegende Entscheidung während der Entwicklung einer Kosimulationsumgebung dar.

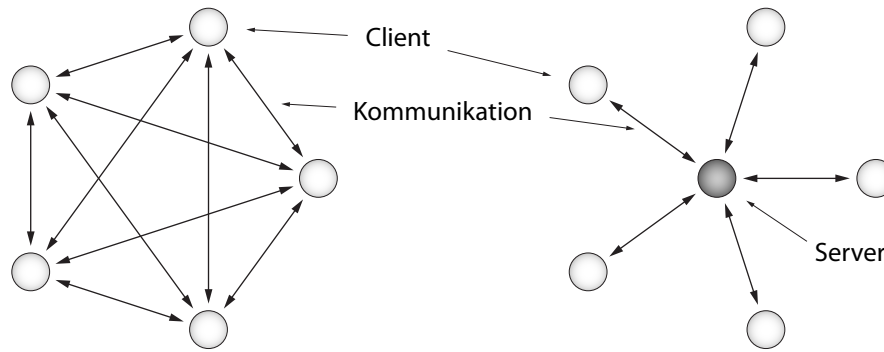


Abbildung 2.6.: Kommunikationsstrukturen nach Lund u. a. [2008]: Direkte Verbindungen (links) und zentraler Server (rechts).

### ♦ Spezialisierte Schnittstellen

Spezialisierte, d.h. an die Kommunikationspartner angepasste Schnittstellen haben den Vorteil, dass die am besten geeignete Technik für die Verbindung gewählt werden kann. Nachteilig wirkt sich aus, dass für jede neue Programmpaarung die Kommunikation angepasst werden muss, wodurch ein System aus mehreren Programmen nicht nur schwer zu erweitern ist, sondern auch sehr wartungsintensiv wird [Lang u. a. 2006].

### ♦ Standardisierte Kommunikation

Wird eine standardisierte Kommunikation eingesetzt, muss zu Beginn der Entwicklung festgelegt werden, über welche Eigenschaften die zu koppelnden Programme verfügen müssen. Hierdurch werden – anders als bei spezialisierten Schnittstellen – evtl. nicht alle von den einzelnen Programmen zur Verfügung gestellten Möglichkeiten genutzt. Demgegenüber steht der Vorteil, dass ein Programm, für das eine Schnittstelle erstellt wurde, ohne weitere Anpassung mit jedem beliebigen anderen Programm kommunizieren kann.

In der Praxis finden sich zumeist Kopplungen von zwei Simulationsprogrammen, die als Solververbund implementiert sind und somit über eine direkte Verbindung miteinander kommunizieren. Dies ist oft auch der erste Schritt im Entwicklungsprozess von Kosimulationen, da die Entwicklung der allgemeinen Elemente wie z.B. des zentralen Servers entfällt. Meistens wird eine auf die eingesetzten Programme spezialisierte Kommunikation eingesetzt, um alle Eigenschaften der Programme nutzen und die Standardisierung umgehen zu können. Ist geplant, eine solche prototypische Umsetzung in den allgemeinen Entwicklungsablauf zu integrieren, ist häufig jedoch die Meinung zu finden, dass die Schnittstellen durch ein einheitliches Tool ersetzt werden sollten (z.B. bei Puntigam u.a. [2006b] und Lund u. a. [2008]). Soll zudem ein Simulationsclient mit mehr als einem anderen Client kommunizieren, erhöht dies den Entwicklungsaufwand der direkten Verbindung erheblich, da vor allem die Synchronisation deutlich komplexer wird.

Auch wenn Arnold u. Schierz [2009] die übliche Anzahl der beteiligten Modelle einer Kopplung mit „zwei, höchstens drei“ angeben, sind bei der Simulation des gesamten thermischen Systems eines Fahrzeugs deutlich mehr Simulationsteilnehmer notwendig. So beschreiben et-



wa Lund u. a. [2008] als Beispiel aus der Praxis eine Kosimulation aus „zwölf oder mehr“ Modellen. Bereits auf Grund dieser hohen Anzahl scheiden spezialisierte Schnittstellen für die Kommunikation aus.

## 2.4. Auswirkung der Kopplung auf die Simulationszeit

Wird die Kopplung im Sinne des parallelen oder verteilten Rechnens dafür verwendet, ein monolithisches Modell auf weitere Hardware-Ressourcen (Prozessoren, Arbeitsspeicher, ...) zu verteilen (vgl. Abbildung 2.1), wird eine Verringerung der Simulationszeit erreicht. Der hierbei mögliche Geschwindigkeitsgewinn ist abhängig vom simulierten System und der Anzahl der eingesetzten Prozessoren. Nach Amdahls Gesetz gibt es für jedes System eine Obergrenze der sinnvollen Anzahl von Prozessoren, da sich mit jedem Prozessor der Geschwindigkeitsgewinn verringert, weil bestimmte Vorgänge nur sequentiell ausgeführt werden können [Amdahl 1967]. Abbildung 2.7 zeigt diesen Sachverhalt beispielhaft für vier Probleme mit unterschiedlich großen parallel rechenbaren Anteilen. Auch wenn Gustafson sein eigenes Gesetz hergeleitet hat [Gustafson 1988], nach dem auch eine superlineare Beschleunigung möglich ist, zeigt Shi [1996], dass beide Gesetze mathematisch identisch sind.

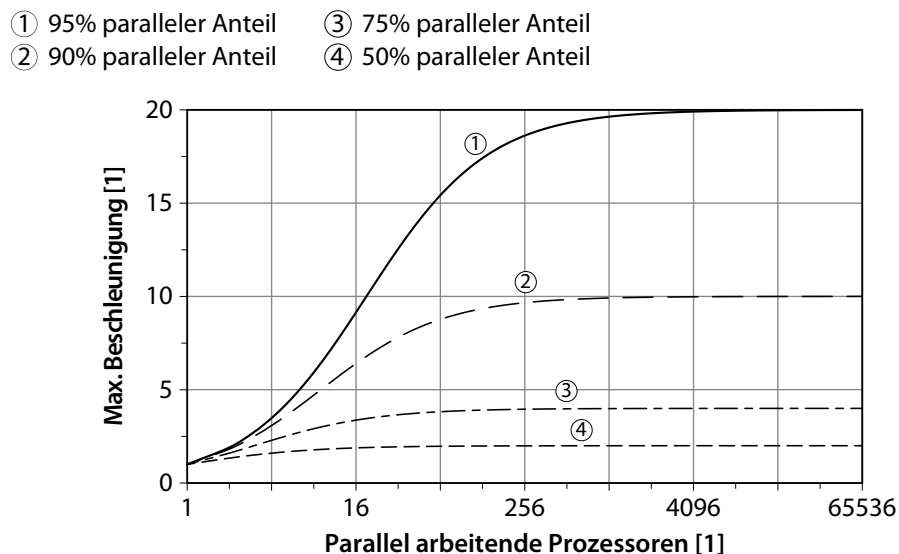


Abbildung 2.7.: Maximale Beschleunigung bei Verwendung parallel rechnender Prozessoren bei der Lösung von Problemen mit unterschiedlichen parallel rechenbaren Anteilen nach Amdahl [1967].

Bei Kosimulationen steht häufig nicht die Simulationszeit im Vordergrund, sondern das Ermöglichen der Verbindung unterschiedlicher Simulationsprogramme, um eine geschlossene Simulation durchführen zu können. Eine Aussage über Geschwindigkeitsvor- oder -nachteile kann im Allgemeinen nicht getroffen werden, da kein Referenzsystem für einen Vergleich zur

Verfügung steht. Wie in Kapitel 2.2.3 beschrieben, hat die parallele Synchronisation das Potential, die Simulationszeit durch gleichzeitiges Ausführen der einzelnen Teilprozesse deutlich zu verringern. Bei jeder Kopplung ist zu beachten, dass durch die Kommunikation und die Synchronisierung immer ein zeitlicher Overhead eingeführt wird [Kim u. a. 2002]. Je größer die Zahl der Teilsysteme und die Abhängigkeiten zwischen den Teilsystemen sind, desto sinnvoller ist der Einsatz einer Kosimulation [Valderrama u. a. 1998], da der Overhead durch die Kosimulation im Vergleich zu dem Simulationsaufwand eines monolithischen Modells in den Hintergrund tritt. Prinzipiell ist ein Rechenzeitgewinn nur dann zu erwarten, wenn der Berechnungsaufwand im langsamen Teilsystem wesentlich größer ist als im schnellen [Gear 1984].

Werden komplexe monolithische Modelle in Teilsysteme zerlegt und diese gekoppelt berechnet, ist ein Vorteil bei der Initialisierung und den ersten Zeitschritten zu beobachten. Dieser Effekt wird in Kapitel 3.4 anhand eines Beispielsystems beschrieben. Abbildung 2.8 zeigt einen Vergleich der Berechnung des monolithischen Modells mit dem Gleichungslöser Dassl [Petzold 1982] und gekoppelten Berechnungen unter Verwendung des Radau- [Hairer u. Wanner 1996] und des Dassl-Lösers. Da die Kosimulationen in dem Beispiel eine konstante Synchronisationschrittweite verwenden und der Rechenaufwand jedes Zeitschritts verglichen mit dem Aufwand der Synchronisation relativ gering ist, ergeben sich die dargestellten linearen Kurvenverläufe. Der Löser der geschlossenen Simulation kann aufgrund der stationären Randbedingungen die Berechnungsschrittweite vergrößern, sobald die zeitlichen Änderungen im System (die durch die nicht optimale Initialisierung auftreten) abgeklungen sind, und somit die Rechengeschwindigkeit erhöhen.

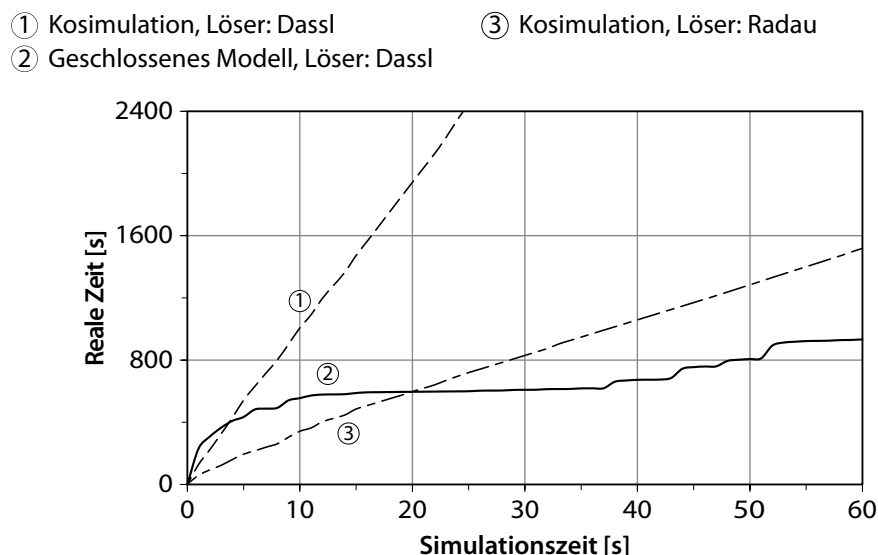


Abbildung 2.8.: Einfluss von Kopplung und gewähltem Gleichungslöser auf die Rechenzeit für das Beispiel aus Kapitel 3.4 bei stationären Randbedingungen.

Wird ein System, dessen Komponenten stark unterschiedliche Zeitkonstanten aufweisen, so

aufgeteilt, dass in jedem Teilsystem nur Komponenten mit jeweils ähnlichen Zeitkonstanten enthalten sind, können die numerischen Löser der einzelnen Teilsysteme mit individuell angepassten Schrittweiten rechnen und bilden somit ein verteiltes Multirate-Lösungsverfahren [Arnold 2006]. Somit ist auch eine Kosimulation mit verschiedenen Modellen desselben Programms sinnvoll, wenn dieses über keinen Multirate-Löser verfügt. Ein Beispiel für ein solches Vorgehen wird in Kossel u.a. [2008] vorgestellt. Hierin wird eine Wärmepumpe für den Einsatz in einem Reisebus simuliert, wobei der Verdichter durch ein sehr detailliertes mechanisches und thermodynamisches Modell beschrieben wird. Für die übrigen Komponenten kommen Modelle aus der Modelica-Bibliothek TIL [Gräber u. a. 2009] zum Einsatz (vgl. Abbildung 2.9). Hierdurch ergeben sich Zeitkonstanten von ca. 0,005 s für den Verdichter und 0,5 s für die übrigen Komponenten. Da die eingesetzte Software Dymola keinen Multirate-Löser zur Verfügung stellt, wurde der Verdichter als eigenes Teilmodell berechnet und über eine Kosimulation mit den übrigen Kreislaukomponenten verbunden. Durch dieses Vorgehen konnte die benötigte Simulationszeit um Faktor 40 gesenkt werden.

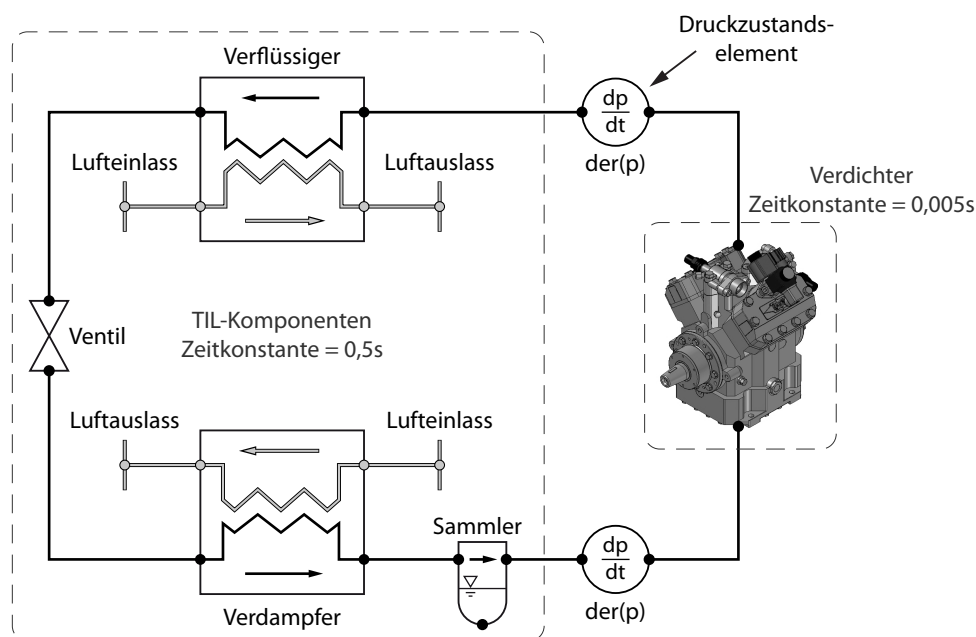


Abbildung 2.9.: Kältekreislaufmodell bestehend aus Komponenten der Bibliothek TIL und einem detaillierten Kompressormodell [Kossel u.a. 2008].



### 3. Behandlung Numerischer Effekte bei der Kopplung

Bei der Kosimulation werden die teilnehmenden Modelle mit einem eigenen Löser berechnet, die hierfür voneinander unabhängige interne Löschrittweiten verwenden können. Hierdurch ergibt sich ein verteiltes Multi-Rate-Verfahren, das das Potential einer erheblichen Geschwindigkeitssteigerung hat [Jackson 1991, Oberschelp u. Vocking 2004]. Negative Auswirkungen auf die benötigte Simulationszeit hat die Synchronisation der ausgetauschten Variablen [Kim u. a. 2002]. Um den genauen Synchronisationszeitpunkt zu finden, wird im Simulationsprogramm ein Zeitereignis ausgelöst und somit der Gleichungslöser gestoppt. Nach erfolgter Datenübertragung wird dieser neu gestartet, wobei das gesamte algebraische Gleichungssystem neu initialisiert werden muss, was eine stark negative Auswirkung auf die Simulationszeit mit sich bringt [Arnold 2004].

Durch das Austauschen von Variablenwerten zu festgelegten Zeitpunkten (vgl. Synchronisationsmethoden in Kapitel 2.2) entstehen auf der Empfangsseite zeitdiskrete, wertkontinuierliche Werteverläufe. Wird der Wert der Variable  $y$  zwischen den einzelnen Synchronisationszeitpunkten konstant gehalten, ergibt sich ein Stufenverlauf, wie ihn Abbildung 3.1 exemplarisch für einen Sinus zeigt.

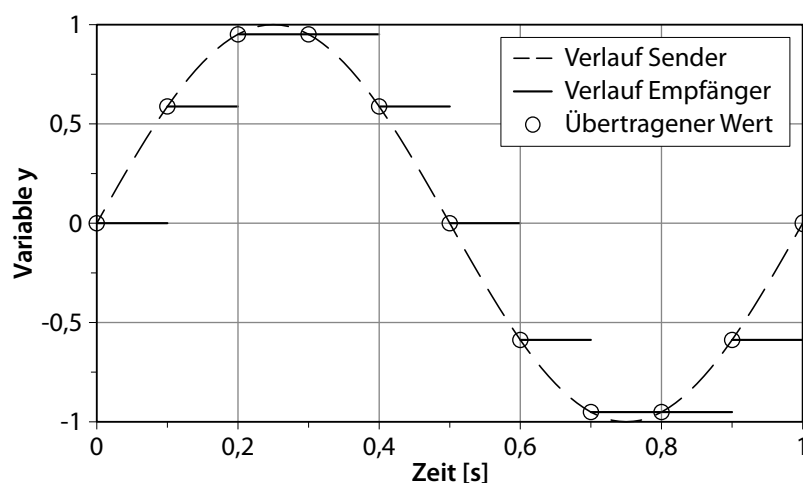


Abbildung 3.1.: Werteverlauf eines Sinussignals auf Sende- und Empfangsseite bei einer Austauschrate von 0,1 s.

Da die zuvor gefundene Lösung als Ausgangspunkt für die Reinitialisierung verwendet wird, erschwert der diskrete Werteverlauf im Empfänger die Reinitialisierung, da sich die Randbedingungen für den algebraischen Lösungsprozess un stetig ändern. Durch eine Verringerung der Synchronisationsrate kann der Unterschied zwischen dem kontinuierlichen Werteverlauf im Sender und dem diskreten Werteverlauf im Empfänger verringert werden. Dieses Vorgehen ist jedoch nur begrenzt zu empfehlen, da hierdurch die Anzahl der Zeitereignisse und somit die für die Simulation benötigte Zeit deutlich erhöht wird. Bei einem zu groß gewählten Austauschintervall kann es hingegen zu großen Sprüngen im empfangenen Werteverlauf kommen; abhängig vom eingesetzten Simulationsprogramm führt dies ggf. zu vergrößerten Rechenzeiten oder einem Abbrechen des Lösungsprozesses [Kübler 2000]. Das Ziel bei der Konfiguration jeder Kosimulation muss es somit sein, die maximale Länge des Synchronisationsintervalls zu bestimmen, mit der eine genaue und gleichzeitig stabile Simulation möglich ist.

In diesem Kapitel werden Möglichkeiten aufgezeigt, das Verhalten der gekoppelten Simulation so zu beeinflussen, dass eine höhere Genauigkeit, Geschwindigkeit und Stabilität erreicht wird. In Kapitel 3.4 werden die vorgestellten Aspekte an einem Beispiel verdeutlicht.

## 3.1. Glättung und Extrapolation empfangener Werte

Wie einleitend ausgeführt, erschwert der zeitdiskrete Werteverlauf der übertragenen Variablen den Lösungsprozess auf der Empfangsseite. Mittels geeigneter Extrapolation können zukünftige Werte abgeschätzt und hierdurch der Sprung im Werteverlauf verkleinert werden. In der Literatur sind mehrere Untersuchungen zum Einsatz von Extrapolation in der Kosimulation zu finden. Kübler [2000] stellt z.B. dar, dass die numerische Stabilität des Systems mit einer höheren Extrapolationsordnung abnimmt und daher die Ordnung auf die übertragene Variable angepasst werden muss.

Das Reinitialisieren wird dem Löser zusätzlich erleichtert, wenn der Werteverlauf und die signifikanten Ableitungen der gekoppelten Variablen stetig sind. Um dies zu erreichen, können Glättungsfunktionen verwendet werden. Da dem Autor keine Literatur bekannt ist, die sich mit einer solchen Art der Glättung beschäftigt, wird diese im Folgenden genauer betrachtet.

Zur Untersuchung von Glättungs- und Extrapolationsfunktionen werden zunächst Bezeichnungen für Funktionen und Punkte eingeführt. Alle Definitionen beziehen sich auf die Seite des Empfängers einer Variablen. Der Werteverlauf der empfangenen Variablen (über den gesamten Zeitverlauf der Simulation) wird als  $F(t)$  bezeichnet. Da sich die folgenden Betrachtungen auf Zeiträume zwischen Synchronisationszeitpunkten beziehen, wird diese Funktion weiter aufgeteilt. Die folgende Nomenklatur ist in Abbildung 3.2 für das Beispiel einer übertragenen Sinus-Funktion dargestellt (vgl. Abbildung 3.1). Als Vereinfachung ist der Werteverlauf der empfangenen Variablen dem eines Haltegliedes (einer Extrapolation 0. Ordnung) nachempfunden. Bei Verwenden einer anderen Extrapolationsmethode wird sich dieser Verlauf ändern.

Als Zählvariable der Synchronisationsschritte wird  $k$  definiert, die Zeitpunkte der einzelnen Synchronisationen werden damit als  $T_k$ , die zugehörigen empfangenen Werte als  $y_k$  bezeichnet. Für den Werteverlauf im Zeitintervall  $T_{k-1} \leq t < T_k$  wird der Name  $F_A(k, t)$  eingeführt, für den Verlauf im Intervall  $T_k \leq t < T_{k+1}$  der Name  $F_N(k, t)$ . Sämtliche weiteren Ausführungen beziehen sich immer auf das Intervall  $T_k \leq t < T_{k+1}$ . Aus diesem Grund wird im Folgenden  $F_N(t)$  an Stelle von  $F_N(k, t)$  und  $F_A(t)$  statt  $F_A(k, t)$  verwendet.

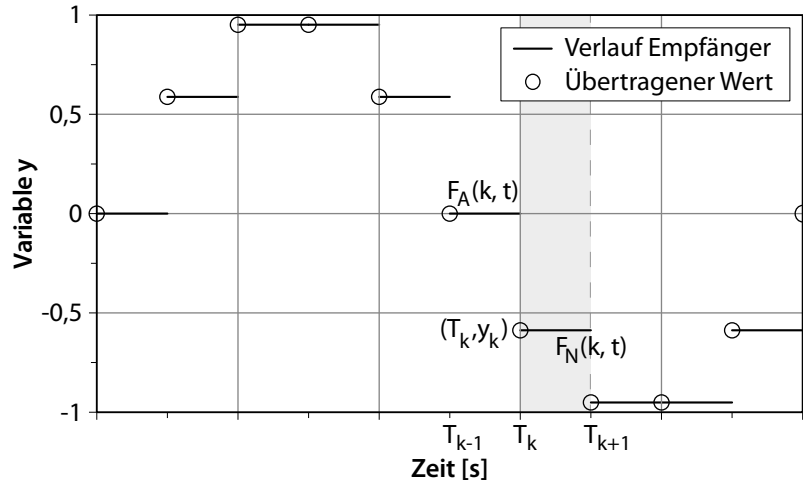


Abbildung 3.2.: Definition der Stützstellen  $(T_k, y_k)$  sowie der Funktionen  $F_A(k, t)$  und  $F_N(k, t)$  im Intervall  $T_k \leq t < T_{k+1}$ .

### 3.1.1. Glättung

Ziel der Glättung ist die Schaffung stetiger Übergänge zu den Synchronisationszeitpunkten  $T_k$ , um die Sprünge im Werteverlauf zu vermeiden und somit einen stetigen Verlauf für  $F(t)$  zu erreichen. Durch Wahl einer geeigneten Glättungsfunktion können auch für die Ableitungen von  $F(t)$  stetige Verläufe geschaffen werden. Im untersuchten Bereich der Simulation thermischer Systeme hat sich die Verwendung einer Glättung als geeignet erwiesen, durch die auch die ersten beiden Ableitungen stetige Verläufe aufweisen.

Um aus  $F_A(t)$  und  $F_N(t)$  den geglätteten Werteverlauf  $F(t)$  zu berechnen, wird die Glättungsfunktion  $\zeta(t)$  eingeführt. Als Parameter für die Glättungsfunktion wird der Umschaltzeitpunkt  $T_s$  verwendet, der aus dem letzten Synchronisationszeitpunkt  $T_k$  und dem nächsten Zeitpunkt  $T_{k+1}$  bestimmt wird. Soll z.B. nach der Hälfte des nächsten Intervalls der Umschaltvorgang beendet sein, ergibt sich:

$$T_s = T_k + 0,5 \cdot (T_{k+1} - T_k) \quad (3.1)$$

Mit der Glättungsfunktion  $\zeta(t)$  wird bis zum Zeitpunkt  $T_s$  von der Funktion  $F_A(t)$  auf  $F_N(t)$

umgeschaltet. Für  $F(t)$  gilt hiermit:

$$F(t) = \begin{cases} \zeta(t) \cdot F_A(t) + (1 - \zeta(t)) \cdot F_N(t) & \text{für } T_k \leq t < T_s \\ F_N(t) & \text{sonst} \end{cases} \quad (3.2)$$

Für die ersten beiden Ableitungen folgt im Bereich  $T_k \leq t < T_s$ :

$$F'(t) = \zeta'(t) (F_A(t) - F_N(t)) + \zeta(t) (F'_A(t) - F'_N(t)) + F'_N(t) \quad (3.3a)$$

$$F''(t) = \zeta''(t) (F_A(t) - F_N(t)) + 2 \cdot \zeta'(t) (F'_A(t) - F'_N(t)) + \zeta(t) (F''_A(t) - F''_N(t)) + F''_N(t) \quad (3.3b)$$

verallgemeinert

$$F^{(n)}(t) = \zeta^{(n)}(t) (F_A(t) - F_N(t)) + \left( \sum_{i=1}^n (n-i-1) \zeta^{(n-i)}(t) (F_A^{(i)}(t) - F_N^{(i)}(t)) \right) + F_N^{(n)}(t) \quad (3.3c)$$

Als Normierung wird  $x(t)$  als dimensionslose Zeit für den Bereich  $T_k \leq t \leq T_s$  definiert:

$$x(t) = \frac{t - T_k}{T_s - T_k} \quad (3.4)$$

Wie oben ausgeführt, können abhängig vom modellierten System und dem gewählten Gleichungslöser die Ableitungen von  $F(t)$  einen erheblichen Einfluss auf die Stabilität und die Geschwindigkeit des Lösungsprozesses haben. Um zu garantieren, dass  $F(t)$  z.B. zweimal stetig differenzierbar ist, muss auch  $\zeta(t)$  zweimal stetig differenzierbar sein. Um sicherzustellen, dass  $F(t)$   $n$ -mal stetig differenzierbar ist, muss gelten:

$$\zeta(x=0) = 1 \quad (3.5a)$$

$$\zeta(x=1) = 0 \quad (3.5b)$$

$$\zeta^{(i)}(x=0) = \zeta^{(i)}(x=1) = 0 \quad \text{für } i = 1, \dots, n \quad (3.5c)$$

Durch Einsetzen dieser Bedingungen in Gleichungen (3.2), (3.3a) und (3.3b) wird die Stetigkeit und stetige Differenzierbarkeit von  $F(t)$  gezeigt. Soll für  $\zeta(t)$  eine Polynomfunktion verwendet werden, ergibt sich für  $n = 2$  das folgende Polynom fünften Grades:

$$\zeta(x) = -6x^5 + 15x^4 - 10x^3 + 1 \quad (3.6a)$$

$$\begin{aligned} \zeta'(x) &= -30x^4 \cdot x' + 60x^3 \cdot x' - 30x^2 \cdot x' \\ &= \frac{1}{T_s - T_k} (-30x^4 + 60x^3 - 30x^2) \end{aligned} \quad (3.6b)$$

$$\zeta''(x) = \frac{1}{T_s - T_k} (-120x^3 \cdot x' + 180x^2 \cdot x' - 60x \cdot x')$$



$$= \frac{1}{(T_s - T_k)^2} (-120x^3 + 180x^2 - 60x) \quad (3.6c)$$

Für das gesamte Zeitintervall  $T_k \leq t < T_{k+1}$  gilt für  $\zeta(t)$ :

$$\zeta(t) = \begin{cases} -6x(t)^5 + 15x(t)^4 - 10x(t)^3 + 1 & \text{für } T_k \leq t < T_s \\ 0 & \text{sonst} \end{cases} \quad (3.7)$$

Abbildung 3.3 zeigt beispielhaft die Verläufe von  $\zeta(t)$ ,  $\zeta'(t)$  und  $\zeta''(t)$  für  $T_s - T_k = 1$  s. Wird  $T_s$  anders gewählt ändert sich der Verlauf von  $\zeta(t)$  nicht, wohl aber die Amplituden der Ableitungen.

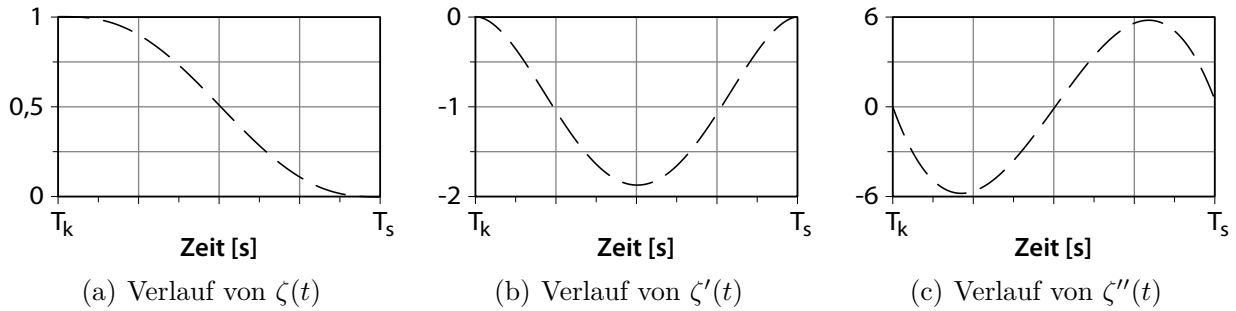


Abbildung 3.3.: Verlauf des Wertes (a) sowie erster (b) und zweiter (c) Ableitung der Glättungsfunktion  $\zeta(t)$  nach Gleichung (3.6) für  $T_s - T_k = 1$  s.

Den Verlauf der geglätteten Funktion  $F(t)$  mit  $T_s$  nach Gleichung (3.1) zeigt Abbildung 3.4.

### 3.1.2. Extrapolation

Bei einer Extrapolation werden Werte abgeschätzt, die außerhalb der verwendeten Stützstellen liegen. Bezogen auf die Kosimulation kann so basierend auf bereits empfangenen Werten prognostiziert werden, welcher Wert zum nächsten Synchronisationszeitpunkt für die empfangene Variable übertragen werden wird. Während der Simulation erfolgt die Berechnung der Extrapolation abschnittsweise, da laufend neue Stützstellen hinzukommen. Hierbei werden nach jedem Synchronisationsschritt auch die Funktionen  $F_A(t)$  und  $F_N(t)$  (vgl. Gleichung (3.2)) neu bestimmt. Für die Simulation thermischer Systeme hat sich im Rahmen dieser Arbeit die Extrapolation mit kubischen Polynomen bewährt. Abbildung 3.5 zeigt exemplarisch die ersten drei kubischen Extrapolationspolynome für ein übertragenes Sinussignal. *Polynom 1* ist dabei aus den ersten vier Stützstellen ( $t = 0$  s bis  $t = 0,3$  s) gebildet, *Polynom 2* aus den Stützstellen 2 bis 5 und *Polynom 3* aus den Stützstellen 3 bis 6.

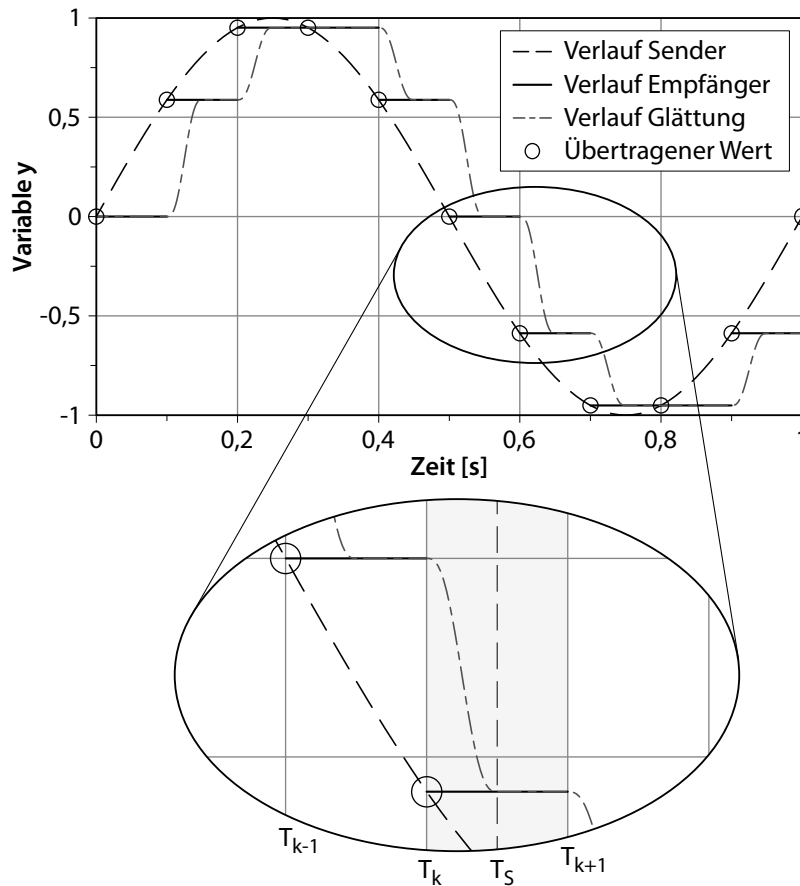


Abbildung 3.4.: Anwendung der Glättung mit vollständiger Umschaltung nach einem halben Synchronisationsintervall ( $T_s = T_k + 0,5 \cdot (T_{k+1} - T_k)$ ).

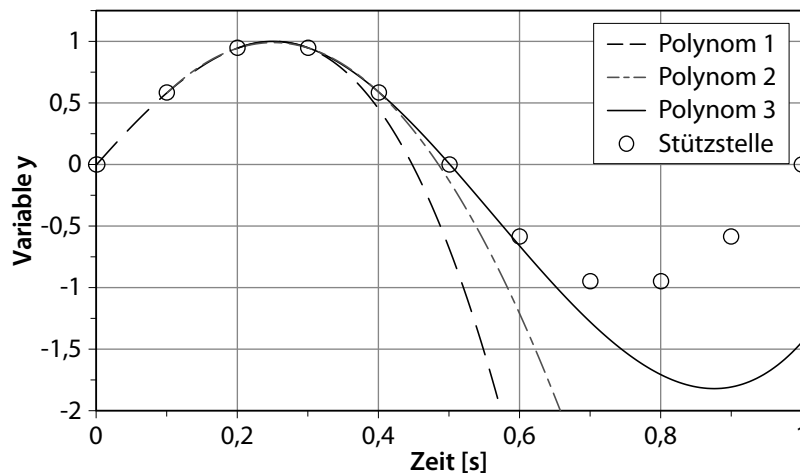


Abbildung 3.5.: Mit 0,1 s abgetastetes Sinussignal und mit jeweils vier Stützstellen gebildete kubische Extrapolationspolynome.

Die Extrapolation birgt sowohl das Potenzial, die Abweichung vom realen Wert zu verringern, als auch das Problem, dass die Abweichung bei ungeeigneter Extrapolation vergrößert wird. Da jede Variable einen individuell unterschiedlichen typischen Verlauf aufweist, muss die Extrapolationsmethode spezifisch für jede Variable konfiguriert werden [vgl. Kübler 2000].

Wird zusätzlich zur Extrapolation mit kubischen Polynomen die in Abbildung 3.4 dargestellte Glättung verwendet, ergibt sich der in Abbildung 3.6 dargestellte Verlauf für  $F(t)$ . Da zum Zeitpunkt  $t = 0\text{ s}$  – dem Simulationsbeginn – nicht ausreichend viele Stützstellen für die Berechnung des Polynoms vorliegen, wird zunächst mit einem Halteglied (Polynom 0. Grades) begonnen und der Grad des Extrapolationspolynoms mit jeder Synchronisierung gesteigert, bis am Zeitpunkt  $t = 0,3\text{ s}$  die benötigten vier Stützstellen für  $F_N(t)$  und zum Zeitpunkt  $t = 0,4\text{ s}$  auch für  $F_A(t)$  empfangen wurden.

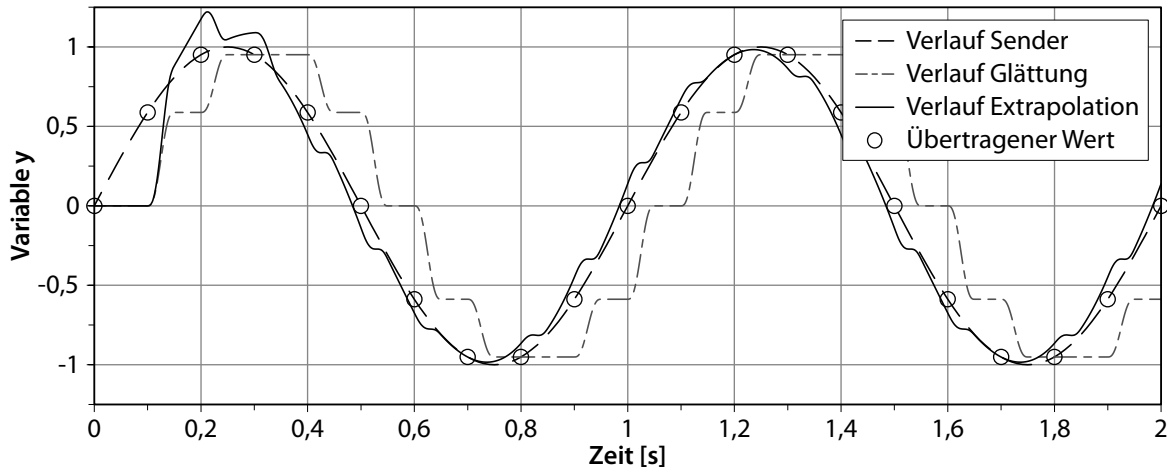


Abbildung 3.6.: Werteverlauf einer Extrapolation mit kubischen Polynomen und polynomieller Glättung fünften Grades.

Um die Abweichung der Werteverläufe von Sender und Empfänger in den ersten Zeitschritten zu verringern, kann zu Beginn der Simulation eine kleinere Synchronisationsrate verwendet werden. In Abbildung 3.7 sind die identischen Werteverläufe wie in Abbildung 3.6 dargestellt, zu Beginn wurde jedoch eine Synchronisationsrate von  $0,02\text{ s}$  eingestellt.

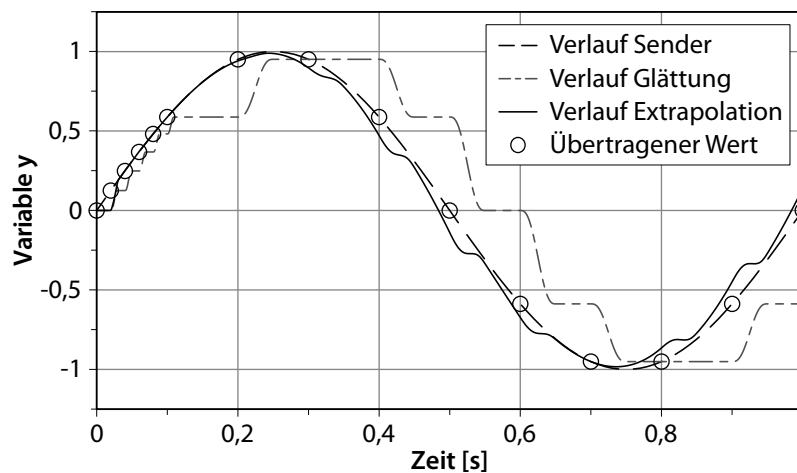


Abbildung 3.7.: Verringerung der Abweichung durch höhere Synchronisationsraten zu Simulationsbeginn (vgl. Abbildung 3.6).

## 3.2. Erhaltung der vollen Dynamik von Randwerten

In transienten Simulationen werden dynamische Randwerte verwendet, um wechselnde Betriebsbedingungen vorzugeben. Im Gegensatz zu den in Modellen berechneten Größen sind die Verläufe der Randwerte bereits vor Beginn der Simulation als Zeitreihe bekannt. Ein Beispiel für eine solche Randbedingungszeitreihe ist ein Fahrprofil, durch das Variablen wie Fahrzeuggeschwindigkeit oder Umgebungstemperatur vorgegeben werden. Oft benötigen mehrere Simulationsmodelle Zugriff auf diese Variablen, so dass es bei einer Kosimulation prinzipiell zwei Möglichkeiten gibt, die Daten bereitzustellen: (1) das Einbinden der Zeitreihen in das Modell jeden Clients und (2) das Bereitstellen der Daten über einen eigenen Client. Sollen die Randbedingungszeitreihe in jeden Client eingebunden werden, muss sichergestellt sein, dass jeder Client identische Informationen verwendet. Je nach Datenmenge und Komplexität der Randbedingung kann die Simulation z.B. durch einen erhöhten Speicher- oder Rechenbedarf erschwert werden. Während bei einer Einbindung in die Modelle die Daten der Randbedingungen für jeden beliebigen Zeitpunkt verfügbar sind, birgt die zentrale Bereitstellung über einen eigenen Client das Problem, dass analog zu den während der Simulation ausgetauschten Variablen die Werte der Zeitreihen nur zu jedem Synchronisationszeitpunkt aktualisiert werden und somit die eigentlich bekannte Dynamik unzureichend genau abgebildet wird.

Da im Gegensatz zu den berechneten Variablen der zukünftige Werteverlauf bekannt ist, können beim Synchronisieren zusätzlich zum aktuellen Wert auch zukünftige Stützstellen übertragen werden. Hierdurch wird es den empfangenden Clients ermöglicht, entweder durch Interpolation dieser Stützstellen die gesamte Dynamik zu rekonstruieren oder wie bei normalen Variablen nur den aktuellen Wert zu verwenden (etwa, wenn der Client die Interpolation nicht unterstützt). Analog zur Extrapolation (vgl. Kapitel 3.1) muss die Art der Interpolation auf die übertragene Variable abgestimmt werden. Wird z.B. der eingelegte Gang in einem Getriebe übertragen, sollte eine Interpolation 0. Ordnung – ein Halten des letzten Wertes – eingestellt werden. Wie bereits bei der Extrapolation erwähnt, hat sich im Rahmen dieser Arbeit die Verwendung kubischer Splines bei thermischen Größen als sinnvoll erwiesen. Da nicht alle Stützstellen zu Beginn der Simulation vorliegen, sondern während der Synchronisation ergänzt werden, bieten sich dynamische Splines als effiziente Alternative an.

Dynamische Splines sind eine Nachbildung kubischer Splines. Anstatt alle Polynomkoeffizienten im Voraus zu berechnen, werden sie während der Simulation und ausschließlich für den jeweils aktuellen Spline-Abschnitt bestimmt. Diese Modifikation des Verfahrens wird z.B. in der Bahnplanung von Robotern verwendet, um den Rechenaufwand auf mehrere Zeitabschnitte zu verteilen [Hesselbach 2003]. Die einzelnen Spline-Abschnitte  $S_k$  werden wie natürliche kubische Splines berechnet. Da in dieser Arbeit eine Stetigkeit des Werteverlaufs und der ersten beiden Ableitungen gefordert wird, muss der Grad des jeweils ersten Spline-Segments ( $S_{k,0}$ ) auf vier erhöht werden, um den stetigen Anschluss an den vorhergehenden Spline-Abschnitt zu gewährleisten. Hierdurch ergibt sich das in Abbildung 3.8 dargestellte Berechnungsschema.

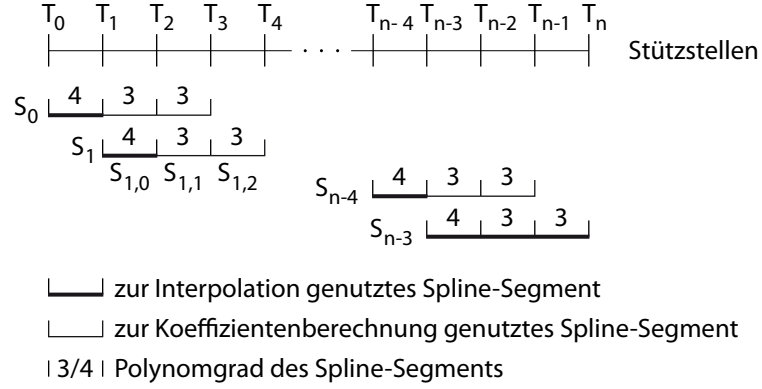


Abbildung 3.8.: Berechnungsschema des dynamischen Splines [nach Hesselbach 2003].

Für die einzelnen Spline-Abschnitte  $S_k$  ( $k = 0, \dots, n-3$ ) gilt somit:

$$S_{k,0}(t) = a_{k,0} + b_{k,0} \cdot x_{k,0}(t) + c_{k,0} \cdot x_{k,0}(t)^2 + d_{k,0} \cdot x_{k,0}(t)^3 + e_{k,0} \cdot x_{k,0}(t)^4 \quad (3.8a)$$

$$S_{k,i}(t) = a_{k,i} + b_{k,i} \cdot x_{k,i}(t) + c_{k,i} \cdot x_{k,i}(t)^2 + d_{k,i} \cdot x_{k,i}(t)^3, \text{ für } i = 1, 2 \quad (3.8b)$$

Hierbei wird die auf die Abschnitte normierte Laufvariable  $x(t)$  verwendet:

$$x_{k,i}(t) = \frac{t - T_{k+i}}{T_{k+i+1} - T_{k+i}} \quad (3.9)$$

Die Anschlussbedingungen für  $k = 1, \dots, n-3$  sind:

$$S_{k,0}(x_{k,0} = 0) = S_{k-1,0}(x_{k-1,0} = 1) \quad (3.10a)$$

$$S'_{k,0}(x_{k,0} = 0) = S'_{k-1,0}(x_{k-1,0} = 1) \quad (3.10b)$$

$$S''_{k,0}(x_{k,0} = 0) = S''_{k-1,0}(x_{k-1,0} = 1) \quad (3.10c)$$

Weiterhin soll gelten:

$$S'_{0,0}(0) = \eta \quad (3.11a)$$

$$S''_{0,0}(0) = \xi \quad (3.11b)$$

$$S''_{k,2}(x_{k,2} = 1) = 0, \text{ für } k = 0, \dots, n-3 \quad (3.11c)$$

Hierbei sind  $\eta$  und  $\xi$  Parameter für die Anfangsbedingungen. Sie entsprechen der ersten bzw. zweiten Ableitung des ersten Segments des ersten Spline-Abschnitts zu Simulationsbeginn. Als Festlegung hat sich bewährt, als Steigung die Gerade zwischen den ersten beiden Stützstellen und keine Krümmung zu verwenden:

$$\eta = \frac{y_1 - y_0}{T_1 - T_0} \quad (3.12)$$

$$\xi = 0 \quad (3.13)$$

Da während der Simulation mit Ausnahme des letzten Spline-Abschnitts nur der Werteverlauf im ersten Segment der Splines berechnet werden muss, ist es ausreichend, ausschließlich die Koeffizienten für dieses Segment zu berechnen. Nach Aufstellen und Umformen des Gleichungssystems ergeben sich diese zu:

$$a_{k,0} = y_k \quad (3.14)$$

$$b_{k,0} = \begin{cases} \eta & \text{für } k = 0 \\ b_{k-1,0} + 2 \cdot c_{k-1,0} + 3 \cdot d_{k-1,0} + 4 \cdot e_{k-1,0} & \text{sonst} \end{cases} \quad (3.15)$$

$$c_{k,0} = \begin{cases} 0,5 \cdot \xi & \text{für } k = 0 \\ c_{k-1,0} + 3 \cdot d_{k-1,0} + 6 \cdot e_{k-1,0} & \text{sonst} \end{cases} \quad (3.16)$$

$$d_{k,0} = \frac{1}{11} \cdot \left( -30 \cdot y_k + 35 \cdot y_{k+1} - 6 \cdot y_{k+2} + y_{k+3} - 26 \cdot \eta - \frac{59}{6} \cdot \xi \right) \quad (3.17)$$

$$e_{k,0} = \frac{1}{11} \cdot \left( 19 \cdot y_k - 25 \cdot y_{k+1} + 6 \cdot y_{k+2} - y_{k+3} + 15 \cdot \eta + \frac{26}{6} \cdot \xi \right) \quad (3.18)$$

Wie in Kapitel 3.1 bezeichnet  $y_k$  hierbei den zum Zeitpunkt  $T_k$  empfangenen Wert.

Zur Überprüfung der Aussage, dass kubische Splines durch dynamische ersetzt werden können, sind in Abbildung 3.9 die Interpolationen zeitdiskreter Stützstellen mit kubischem und dynamischem Spline aufgetragen. Die zusätzlich dargestellte Abweichung ist auf den Verlauf des kubischen Splines bezogen.

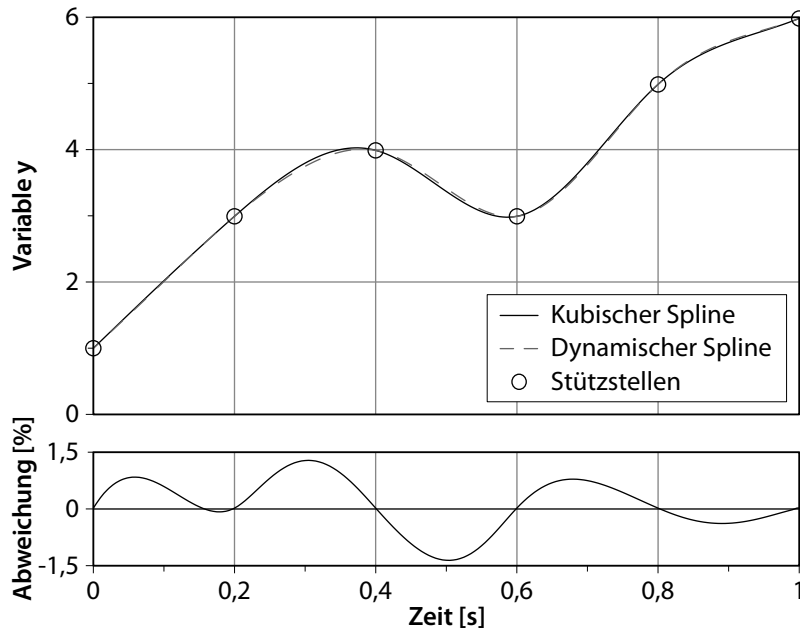


Abbildung 3.9.: Vergleich von Interpolationen mit kubischem und dynamischem Spline, Abweichung bezogen auf den Wert des kubischen Splines.

Mit speziellen Simulationsclients können abhängige Randbedingungszeitreihen berechnet werden. Voraussetzung hierfür ist, dass diese Berechnungen ausschließlich auf anderen Randbe-

dingungszeitreihen beruhen, also keine weiteren Variablen empfangen werden. Abbildung 3.10 zeigt ein Beispiel für eine Kosimulation zur Klimatisierung eines Innenraums im Umluftbetrieb. Hierbei werden durch ein Fahrprofil Randbedingungszeitreihen für die Position des Fahrzeugs und die Motordrehzahl bereitgestellt. Aus der Position werden die klimatischen Bedingungen bestimmt, die als abhängige Randbedingungszeitreihen an die weiteren Simulationsclients geschickt werden.

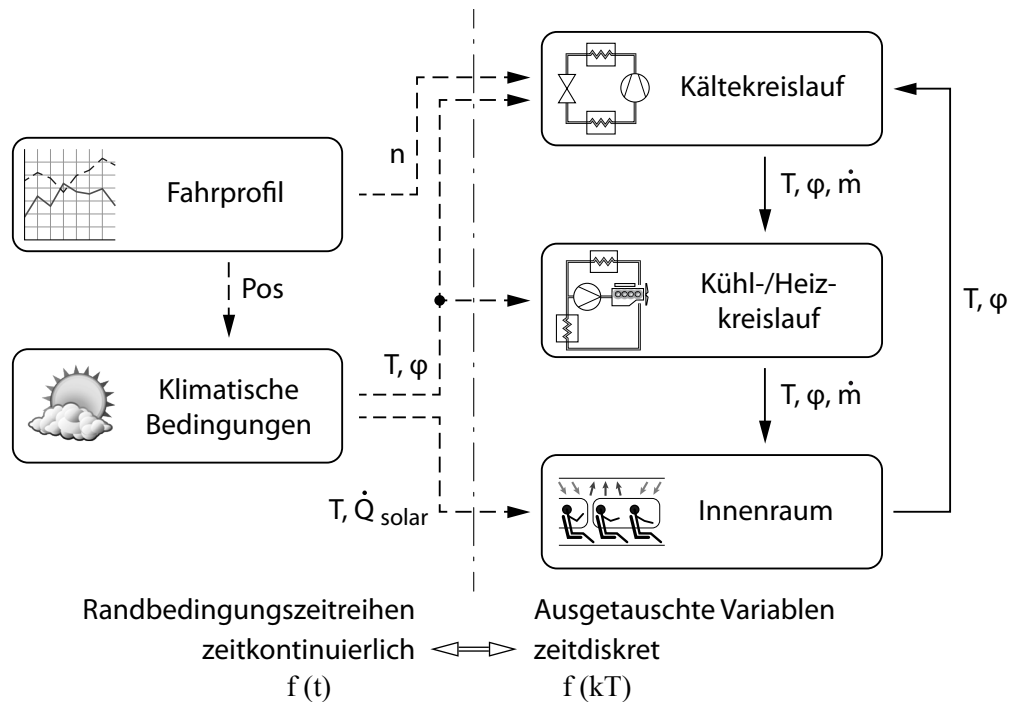


Abbildung 3.10.: Modellschema einer Kosimulation mit Randbedingungszeitreihen.

Für die abhängigen Randbedingungszeitreihen wird in der Synchronisation ein Zwischenschritt vorgesehen (vgl. Abbildung 3.11). Hierbei wird zunächst ausschließlich die Kommunikation zwischen der Quelle der Randbedingungszeitreihe (im Beispiel das Fahrprofil) und der Berechnung der abhängigen Zeitreihe – den klimatischen Daten – durchgeführt. Im Anschluss hieran erfolgt die übrige Synchronisation wie gewohnt.

### 3.3. Auswahl und Behandlung auszutauschender Variablen

Die Auswahl der zwischen den verschiedenen Teilsystemen auszutauschenden Variablen ist stark davon abhängig, welcher technischen Disziplin die Systeme entstammen und welche Größen in den verschiedenen Teilsystemen zur Verfügung stehen. Da es somit keine Universallösung für die Auswahl der Variablen geben kann, werden hier lediglich Teilaspekte betrachtet.

Unabhängig von der technischen Disziplin können Variablen unterschieden werden in Zustandsgrößen, die ausschließlich vom Zustand des Systems abhängig sind, nicht aber vom

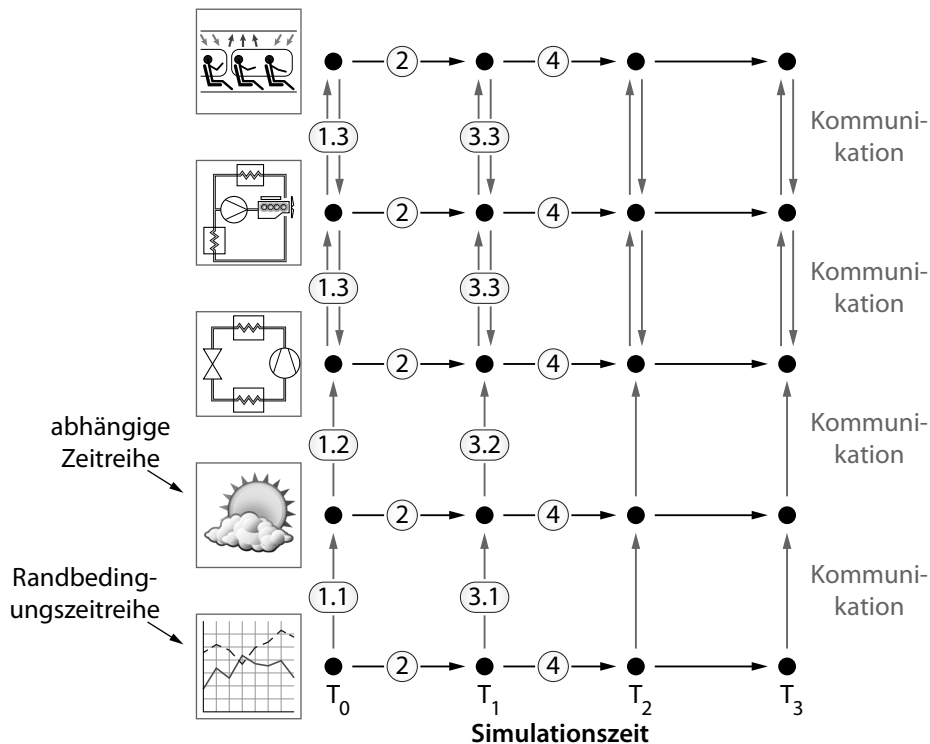


Abbildung 3.11.: Parallele Synchronisation mit gesonderter Behandlung von Zeitreihen (System nach Abbildung 3.10, Kommunikation vereinfacht dargestellt).

Prozess, mit dem das System diesen Zustand erlangt hat, und in Prozessgrößen, die auch vom Prozess der Systemänderung abhängen [Weigand u. a. 2008]. Tabelle 3.1 zeigt für ausgesuchte technische Disziplinen die jeweils zugeordneten Zustands- und Prozessgrößen.

Techn. Disziplin	Zustandsgröße	Prozessgröße	Energieträger
Elektronik	Spannung	Elektr. Strom	Ladung
1-D Translation	Position	Kraft	Impuls
1-D Rotation	Winkel	Drehmoment	Drehimpuls
Hydraulik	Druck	Volumenstrom	Volumen
Thermodynamik	Temperatur	Wärmestrom	Wärme
Chemie	Chem. Potential	Partikelstrom	Partikel

Tabelle 3.1.: Energieträger mit zugeordneten Zustands- und Prozessgrößen für ausgesuchte technische Disziplinen [nach Fritzson 2004].

Wird ein technisches System aufgeschnitten, müssen die an den Schnittstellen auftretenden Zustands- und die zugehörigen Prozessgrößen zwischen den Teilsystemen ausgetauscht werden. Hierbei wird die Zustandsgröße entgegengesetzt der Prozessgröße übertragen (z.B. die Temperatur von System A zu System B, der Wärmestrom von System B zu System A). Für die Festlegung der Austauschrichtung ist von Relevanz, in welchem Teilsystem die Zustandsgrößen aus Tabelle 3.1 als differentielle Variable verwendet wird. Werden Werte differentieller



Variablen von außen aufgeprägt, führt dies zu einem nichtlinearen Gleichungssystem höherer Ordnung [Arnold 1998]. Zur Vermeidung einer Indexreduktion ist somit die Zustandsvariable aus dem System zu senden und die Prozessgröße zu empfangen. Differentielle Variablen treten allgemein bei Speichertermen auf. Zur Festlegung der Austauschrichtung gilt es somit, zunächst die Speicherterme zu identifizieren – bei thermischen Systemen sind dies thermische Kapazitäten. Für sie gilt mit der Wärmekapazität  $C$  für Festkörper die Gleichung

$$C \cdot \frac{dT}{dt} = \dot{Q} \quad (3.19)$$

Hierbei tritt die Temperatur  $T$  als differentielle Zustandsgröße auf, die dementsprechend von der Kapazität weg in das angekoppelte System übertragen wird - in Abbildung 3.12 vom Teilsystem B zum Teilsystem A. Der Wärmestrom  $\dot{Q}$  stellt die Prozessgröße dar und wird entgegen der Temperatur von Teilsystem A zum Teilsystem B übertragen.

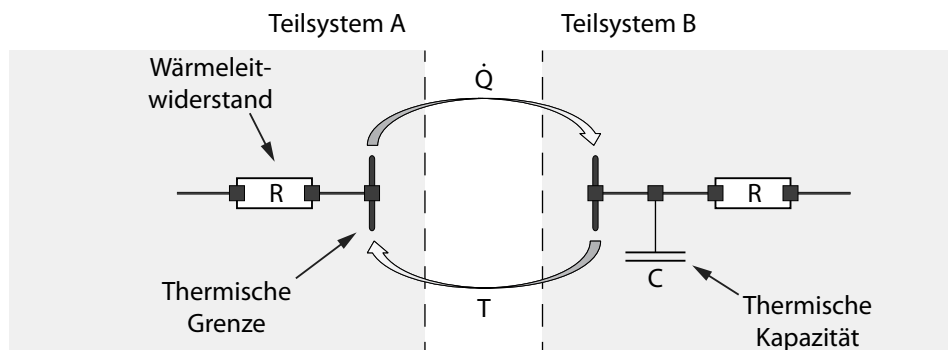


Abbildung 3.12.: In Teilsysteme A und B aufgetrenntes thermisches System mit differentieller Zustandsgröße  $T$  und Prozessgröße  $\dot{Q}$ .

Der Ort des differentiellen Zustands sollte nach Möglichkeit direkt an der Systemgrenze liegen. Bei der Modellierung von Festkörpern wie z.B. Rohrwänden kann dies dadurch erreicht werden, dass bei Aufteilung in Wärmeleitwiderstand und Wärmekapazität die Kapazität an den Rand der Rohrwand gelegt wird. Alternativ kann ein zusätzlicher Speicherterm an der Systemgrenze hinzugefügt werden. Zu beachten ist bei diesem Vorgehen, dass das Systemverhalten geändert wird. Durch Abstimmung der zusätzlich eingebrachten Zeitkonstante auf die durch die Simulation zu untersuchenden Effekte kann dieser Einfluss jedoch soweit gemindert werden, dass er nicht weiter relevant ist.

Sind zur Bestimmung des Sytemzustands weitere Zustandsgrößen notwendig, gilt als Regel, dass extensive Zustandsgrößen mit den zugehörigen Prozessgrößen übertragen werden. Zur Bestimmung des thermischen Zustands von Kältemittel können z.B. der Druck als intensive Zustandsgröße und die Enthalpie (meistens als spezifische Größe) als extensive Zustandsgröße verwendet werden. Bei einem strömenden Medium tritt nach Tabelle 3.1 der Volumenstrom

als Prozessgröße auf. Abbildung 3.13 zeigt die Richtung der ausgetauschten Variablen für die Auftrennung des Kältemittel-Fluidstroms zwischen einem Rohr und einem Ventil.

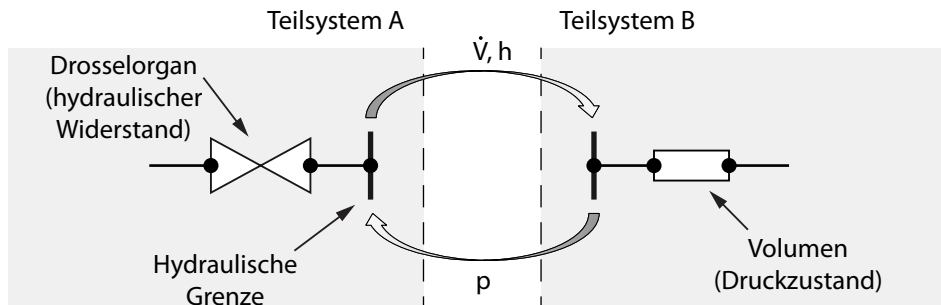


Abbildung 3.13.: In Teilsysteme A und B aufgetrenntes hydraulisches, von Kältemittel durchströmtes System mit differentieller Zustandsgröße  $p$ , zugeordneter Prozessgröße  $\dot{V}$  und spezifischer Zustandsgröße  $h$ .

Wie zu Beginn von Kapitel 3 dargelegt, ist das Ziel bei der Wahl des Synchronisationsintervalls, dieses zu maximieren, um die für die Durchführung der Kosimulation benötigte Zeit zu minimieren. Hierbei ist zu beachten, dass die Genauigkeit der Simulation höher ist, je enger die Modelle gekoppelt sind – mit einem größeren Synchronisationsintervall sinkt die Genauigkeit. Als allgemeine Regel gilt, dass die Länge des Austauschintervalls nicht größer sein darf als die Zeitkonstanten der gekoppelten Systeme, die mit den übertragenen Variablen assoziiert sind. Die kleinste Zeitkonstante der gesamten Kosimulation bestimmt deshalb die Austauschrate sämtlicher Teilmodelle [Trčka 2008]. Durch die in den vorangegangenen Abschnitten dieses Kapitels vorgestellten Methoden der Extrapolation und Glättung kann die Abweichung der Wertnachbildung der übertragenen Variablen auf der Empfangsseite verringert und damit das Synchronisationsintervall vergrößert werden.

Weisen verschiedene Teilmodelle einer Kosimulation deutlich unterschiedliche Zeitkonstanten auf, kann es sinnvoll sein, unterschiedlich große Synchronisationsintervalle für die verschiedenen Modelle zu verwenden. Dies wird in dieser Arbeit durch eine Erweiterung des parallelen Synchronisationsverfahrens (vgl. Kapitel 2.2.1) ermöglicht. Abbildung 3.14 zeigt eine Konfiguration von drei Clients, wobei Client 1 eine doppelt so große Synchronisationsschrittweite verwendet wie die Clients 2 und 3.

Durch Anpassen der Synchronisationsschrittweiten an die Zeitkonstanten der einzelnen Teilsysteme können Systemverhalten und Simulationsgeschwindigkeit optimiert werden. So können beispielsweise aufwändige Modelle mit großen Zeitkonstanten mit einer großen Schrittweite an Modelle mit kleinen Zeitkonstanten gekoppelt werden.

Während einer Kosimulation kann durch einen externen Schrittweitencontroller die Synchronisationsrate in Abhängigkeit vom aktuellen Zustand der Teilmodelle angepasst werden. Ein solcher Schrittweitencontroller kann der Kosimulation z.B. als Clientprogramm hinzugefügt werden. Durch ein Analysieren der ausgetauschten Werte kann bestimmt werden, wie groß

die Änderungen im Gesamtsystem zu jedem Zeitpunkt sind. Mit dieser Information kann in Phasen hoher Dynamik das Austauschintervall verkleinert, in stationären Phasen erhöht werden. Voraussetzung für dieses Vorgehen ist, dass die gekoppelten Simulationswerkzeuge eine Änderung der Synchronisationsrate während einer Simulation unterstützen – ist dies nicht für alle Programme gegeben, kann evtl. auch durch das Anpassen der Austauschrate für einen Teil der Clients ein Geschwindigkeitsgewinn erzielt werden. Durch die Verwendung einer Zeitreihe als zentrale Vorgabe der Randbedingungen (vgl. Kapitel 3.2) kann zudem vermieden werden, den Beginn einer Phase erhöhter Dynamik zu verfehlen.

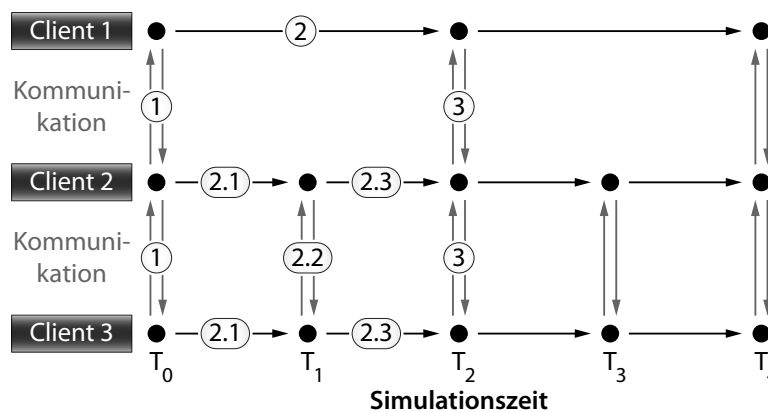


Abbildung 3.14.: Parallele Synchronisation (vgl. Abbildung 2.2), erweitert um die Option unterschiedlicher Synchronisationsraten (Kommunikation vereinfacht dargestellt).

## 3.4. Kopplungseffekte bei Systemsimulationen

Um die Effekte untersuchen und beurteilen zu können, die durch den Einsatz von Kosimulation bei Systemsimulationen auftreten, wird in diesem Kapitel eine Klimaanlage mit dem Kältemittel R-134a betrachtet, wie sie in heutigen PKW zum Einsatz kommt. Dieses System wird gewählt, da Anzahl und Verschaltung der Komponenten gut überschaubar sind und ausreichend Messdaten zur Validierung der Simulationsmodelle zur Verfügung stehen.

Für die Erstellung sämtlicher Modelle wird die Modellierungssprache Modelica [Modelica 2009] sowie die Modellbibliothek TIL [Gräber u. a. 2009] verwendet. Hierdurch kann nicht nur Einfluss auf die Komponentenmodelle genommen werden, um Details näher betrachten zu können, es ist außerdem möglich, eine geschlossene Simulation mit einer Kosimulation zu vergleichen.

### 3.4.1. Betrachtetes System

Die untersuchte PKW-Klimaanlage besteht aus den Komponenten Taumelscheibenverdichter, IRDC (integrierte Komponente bestehend aus Sammler, Trockner und Verflüssiger, englisch

Integrated Receiver Dryer Condenser), IHX (interner Wärmeübertrager), Ventil und Verdampfer. Als Ventil kommt ein TXV (thermostatisches Expansionsventil) zum Einsatz, welches den Öffnungsquerschnitt abhängig von der Kältemitteltemperatur nach dem Verdampfer einstellt. Abbildung 3.15 zeigt den Aufbau des Kreislaufs inklusive der Verrohrung, die dem Aufbau des Prüfstands entspricht (für die Validierung ist die zu den Messdaten passende Verschaltung relevant und nicht die Verschaltung im PKW).

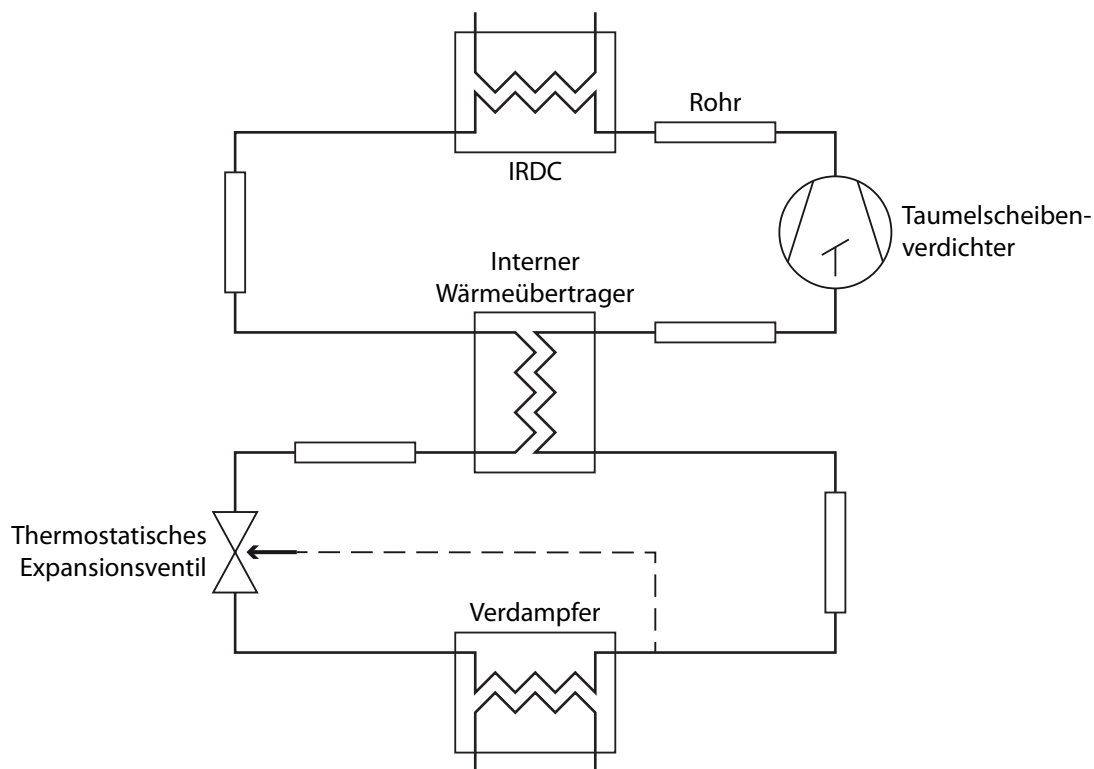


Abbildung 3.15.: Schema des untersuchten R-134a PKW-Klimakreislaufs mit Verrohrung des Prüfstands.

#### Komponentenmodelle

Der Taumelscheibenverdichter wird durch ein kennfeldbasiertes Modell abgebildet, das auf Messungen in Teil- und Volllast beruht [siehe Försterling 2004]. IRDC und Verdampfer sind Flachrohrwärmeübertrager, die für diese Untersuchung durch Modelle mit hohem Detaillierungsgrad abgebildet werden [siehe Strupp u.a. 2009]. Während der IRDC vier Pässe in einer Ebene aufweist, verfügt der Verdampfer über zwei Ebenen, die hintereinander in jeweils mehreren Pässen durchströmt werden. Das TXV berücksichtigt die kompressible Strömung durch eine Kombination von Gleichungen nach Bernoulli und choked-flow-Gleichungen, der Strömungsquerschnitt wird von einem Modell des Thermokopfes gesteuert. Der interne Wärmeübertrager ist ein Rohr-in-Rohr-Wärmeübertrager. Auf Grund der beidseitig hohen Wärmeübergangskoeffizienten und der dünnen, die Kältemittelströme trennenden Wand bewirkt der interne Wärmeübertrager eine enge Kopplung der Kältemittelzustände auf Hoch- und Niederdruckseite.

Die einzelnen Komponentenmodelle werden in Testumgebungen mit Messdaten validiert. Dieses Vorgehen wird im Folgenden exemplarisch für den IRDC dargestellt.

Die Modellierung des IRDC erfolgt nach Strupp u.a. [2009]. Hierbei werden die einzelnen Pässe mit mehreren Kontrollvolumina diskretisiert, so dass die Zustandsgrößen örtlich aufgelöst vorliegen. Jedes Kontrollvolumen verbindet Modelle für den Kältemittel- und den Luftstrom sowie die trennende Wand. Durch die Auswahl geeigneter Transportbeziehungen werden Wärmeübergang und Druckabfall abgebildet. Durch Verwendung unterschiedlich großer Kontrollvolumina kann die Diskretisierung an die Dynamik lokaler Phänomene angepasst werden (vgl. Abbildung 3.16). Insbesondere an Orten, an denen das Kältemittel vom einphasigen in den zweiphasigen Zustand wechselt, sollten auf Grund der ausgeprägten lokalen Änderung von Wärmeübergangskoeffizient und Kältemitteldichte kleinere Kontrollvolumina verwendet werden. Für die Header kommen dieselben Kontrollvolumina zum Einsatz, um neben den auftretenden Druckabfällen auch die enthaltene Kältemittelmasse zu berücksichtigen – eine Wärmeübertragung findet nicht statt. Der integrierte Sammler wird durch ein unabhängiges Kontrollvolumen repräsentiert. Die Trennung der Phasen wird durch eine charakteristische Kurve abgebildet, die einen Zusammenhang zwischen Füllstand und Austrittsdampfgehalt herstellt.

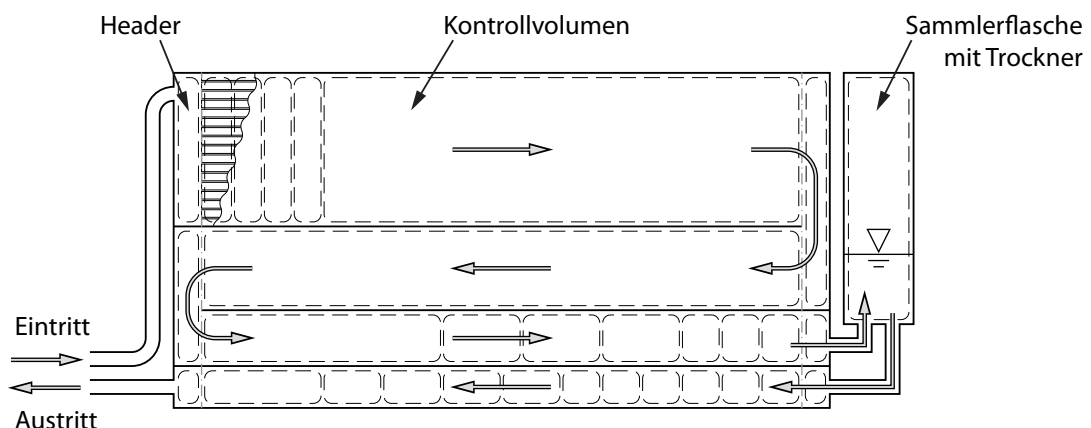


Abbildung 3.16.: Modellierungsschema des IRDC (integrierte Komponente aus Sammler, Trockner und Verflüssiger) [Strupp u.a. 2009].

## Validierung

Die Verifizierung und Validierung von Simulationsmodellen sollte einen wichtigen Punkt in jedem Softwareentwicklungsprozess einnehmen [Sargent u. a. 2006]. Der Schritt der Verifizierung – der Überprüfung, dass das konzeptionelle Systemmodell korrekt umgesetzt ist und dass das Simulationsprogramm korrekt arbeitet [Sargent 2005] – wird im Rahmen dieser Arbeit nicht diskutiert. Es wird davon ausgegangen, dass die Modelle korrekt umgesetzt sind und dass sich die eingesetzte Modellierungssprache Modelica [Modelica 2009] sowie das Simulationsprogramm Dymola [Dymola 2010] wie vorgesehen verhalten.

Nach Lang u. a. [2006] ist es sinnvoll, statt des zusammengeführten Gesamtmodells die Einzelmodelle des Gesamtsystems mit Messdaten abzustimmen, weil die Ursache für Abweichungen um so schwerer gefunden und einem einzelnen Modell zugeordnet werden kann, je höher die Systemkomplexität ist. Hierbei beziehen sich Lang u. a. auf Teilsysteme einer Gesamtfahrzeugsimulation, die Argumentation ist aber auf jedes simulierte System übertragbar. Aus diesem Grund werden hier die Komponentenmodelle einzeln validiert. Die Daten für die Validierung wurden aus Messdaten des Gesamtkreislaufs extrahiert, wodurch sichergestellt wird, dass die Validierung aller Komponenten mit konsistenten und realistischen Einsatzbedingungen durchgeführt wird.

Für die Komponentenvalidierung werden einzelne Testumgebungen erstellt, die sämtliche am Prüfstand vorhandenen Bauteile einschließen. Neben dem IRDC ist somit auch ein Rohr in der Testumgebung vorhanden, da die Messdaten vor der Zuleitung und nach dem IRDC aufgenommen wurden (vgl. Abbildung 3.17). Zusätzlich sind Objekte aus der TIL-Bibliothek enthalten, die der numerischen Effizienz und der Auswertung dienen: Das Druckzustandselement repräsentiert ein Druckniveau – in diesem Fall den Hochdruck. Für dieses Druckniveau wird die zeitliche Änderung des Drucks an jeder Stelle (z.B. in den Kontrollvolumina) gleich gesetzt. Hierdurch kann die Anzahl der Zustandsvariablen bei vernachlässigbarer Abweichung deutlich verringert werden [Lemke 2004]. Durch die ebenfalls hinzugefügten ph-Sensoren kann der IRDC sehr einfach in thermischen Phasendiagrammen dargestellt werden.

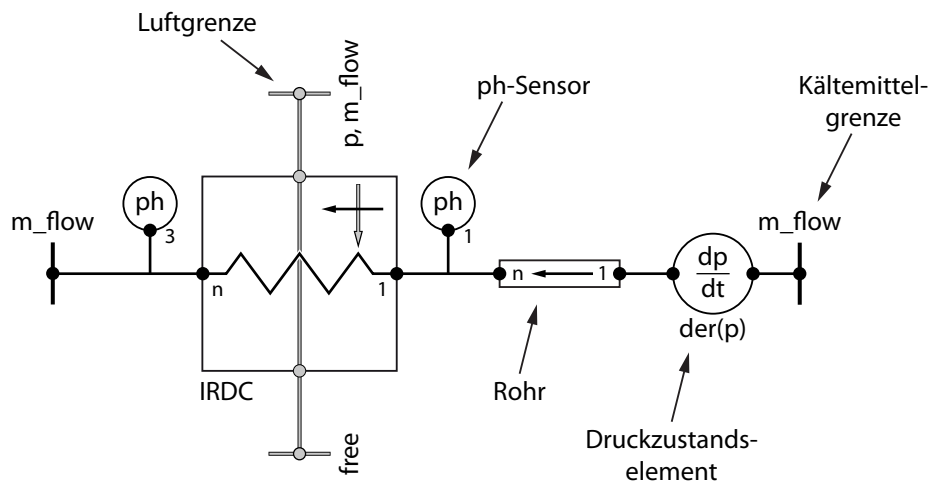


Abbildung 3.17.: Testumgebung zur Validierung des IRDC-Modells.

Für die Validierung werden folgende Schritte nach Sargent [1988] durchgeführt:

- ♦ Überprüfen der Messdaten auf Plausibilität
- ♦ Grafischer Vergleich von Mess- und Simulationsdaten
- ♦ Numerischer Vergleich von Mess- und Simulationsdaten
- ♦ Beaufschlagung der Modelle mit Extrembedingungen

Zusätzlich wird bei jeder Modellanpassung mit Expertenwissen subjektiv überprüft, ob die erzeugten Ergebnisse schlüssig sind [Sargent 2005].

Durch die Überprüfung der Messdaten auf Plausibilität werden solche Punkte aussortiert, die für den Zweck der Validierung nicht geeignet sind. Dieser Schritt sollte am Anfang jeder Modellvalidierung stehen, da diese häufig daran scheitert, dass keine oder nicht ausreichend viele angemessenen und präzisen Daten vorliegen [Sargent 2008]. Die Erkennung von korrekten Daten erfordert eine sorgfältig geplante Vorgehensweise mit einem festgelegten Kriterienkatalog. So können etwa Messpunkte gestrichen werden, bei denen offensichtliche Messfehler aufgetreten sind oder der stationäre Zustand noch nicht erreicht wurde.

Durch einen grafischen Vergleich der Simulationsergebnisse mit Messdaten kann ohne viel Arbeitsaufwand bereits eine erste Aussage zur Validität des Modells getroffen werden. Hierbei werden für diverse Randbedingungen die Simulationsergebnisse in einem Diagramm mit Messdaten aufgetragen und rein optisch bewertet. Ein solches Diagramm ist in der später folgenden Abbildung 3.21 mit einem thermischen Phasendiagramm dargestellt.

Den numerischen Vergleich von gemessenen und simulierten Wärmeströmen am IRDC zeigt Abbildung 3.18. Da sämtliche Punkte innerhalb eines  $\pm 10\%$  Konfidenzintervalls um die gemessenen Werte liegen – bei einer thermischen Effizienz zwischen 65% und 100% –, kann von einem für stationäre Bedingungen validierten Modell ausgegangen werden.

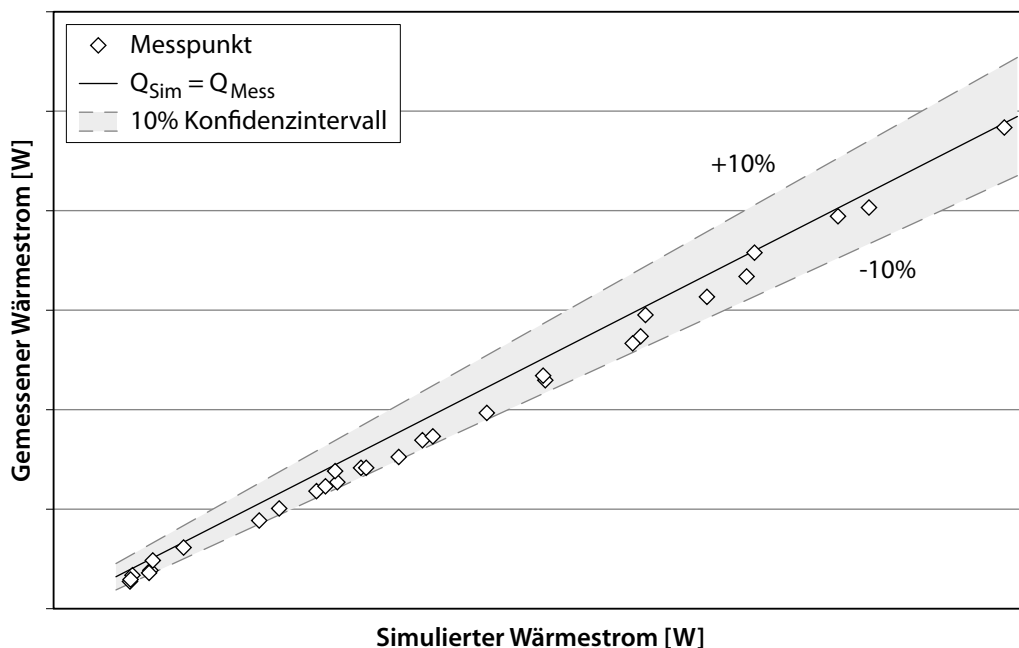


Abbildung 3.18.: Vergleich von gemessenen und simulierten Wärmeströmen des IRDC.

Um die Gültigkeit der Modelle an Betriebspunkten, für die keine Messdaten vorliegen, (die Extrapolationsfähigkeit der Modelle) zu überprüfen, ist das System mit extremen Randbe-

dingungen zu beaufschlagen. Durch Bewertung der Ergebnisse durch einen Experten kann das korrekte Verhalten der Komponentenmodelle getestet und evtl. eine Einschränkung des Einsatzbereiches für das Modell vorgenommen werden [Sargent 2007].

#### Gesamtsystem für die Kosimulation

Nach der Validierung sämtlicher Komponentenmodelle werden zwei Gesamtsystemmodelle erstellt: Ein geschlossenes Modell, das sämtliche Komponenten beinhaltet, und eine Kosimulation, für die eine Aufteilung in Teilsysteme wie folgt vorgenommen wird: Um die zwei Druckniveaus geschlossen simulieren zu können, wird das Modell des internen Wärmeübertragers in zwei Teile aufgesplittet. Verdichter und Ventil werden in eigenen Systemen berechnet, so dass sich die in Abbildung 3.19 gezeigte Aufteilung ergibt. Um die verschiedenen Modelle vergleichen zu können, wird bei allen Simulationen eine identische Parametrierung verwendet.

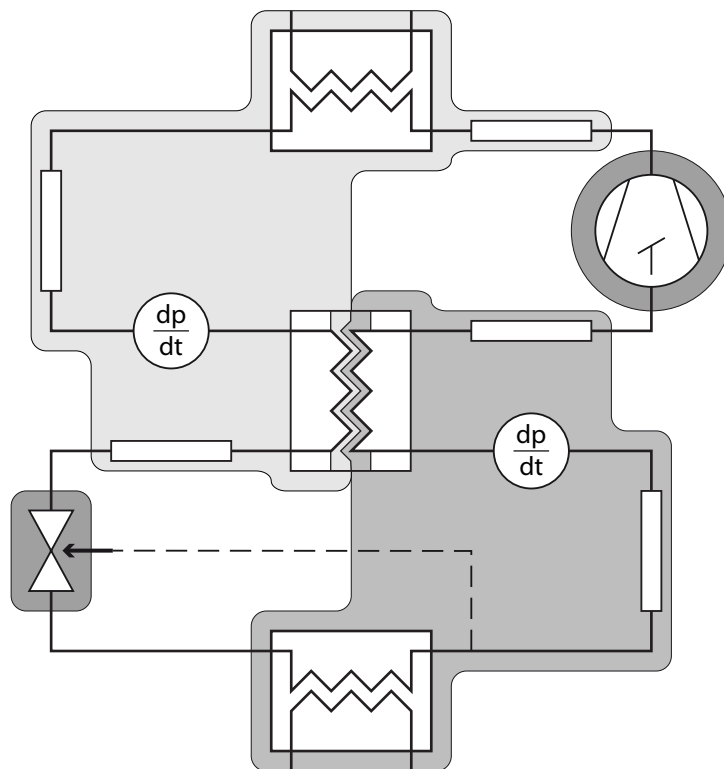


Abbildung 3.19.: Aufteilung des Gesamtsystems nach Abbildung 3.15 in Teilsysteme für die Kosimulation.

Die vier farblich hinterlegten Bereiche kennzeichnen die vier Teilsysteme, die in eigenen Instanzen des Simulationsprogramms Dymola berechnet werden. Die Auftrennung des Modells für den internen Wärmeübertrager ist einfach, da es aus zwei Rohren besteht, deren Wände über Temperatur und Wärmestrom miteinander verbunden sind. Diese Verbindung kann aufgetrennt und durch eine Kommunikation über die Kosimulationsschnittstelle ersetzt werden.



Nach Kapitel 3.3 hängt die maximale Größe der Kopplungsschrittweite von den Zeitkonstanten der zu verbindenden Systeme ab. Hierbei sind sowohl die Zeitkonstanten der Fluidströme wichtig (wie lange braucht ein Fluidpartikel, um die betrachtete Komponente zu durchströmen), als auch die der Wärmeströme. Wie bereits oben beschrieben, weist der interne Wärmeübertrager eine dünne Wand sowie für beide Kältemittelströmungen hohe Wärmeübergangskoeffizienten auf. Diese kleinen Widerstände und Kapazitäten führen zu einer kleinen Zeitkonstante bei der Wärmeübertragung. Im Rahmen dieser Arbeit liegt diese Zeitkonstante nicht im Rahmen der interessierenden Größenordnungen, weshalb zusätzliche Wärmeleitwiderstände und thermische Kapazitäten eingefügt werden, um die Zeitkonstante zu vergrößern (siehe Abbildung 3.20). Hierdurch wird zusätzlich die Bedingung aus Kapitel 3.3 erfüllt, nach der die Werte der Zustandsvariablen übertragen werden sollen: Die Zustandsvariable Temperatur wird vom neu eingeführten Zustand (der Kapazität) an das andere Modell gesendet.

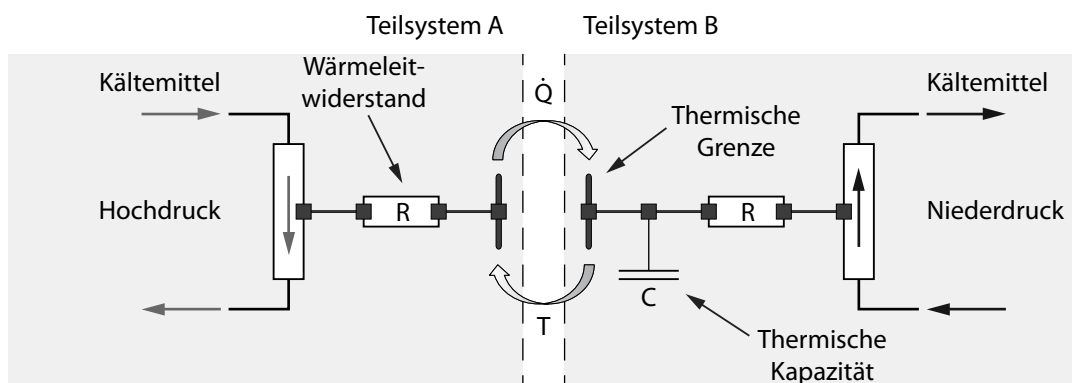


Abbildung 3.20.: Für die Kosimulation modifiziertes Modell des internen Wärmeübertragers mit den zwischen den Grenzen übertragenen Größen Temperatur  $T$  und Wärmestrom  $\dot{Q}$ .

Die Auswirkungen dieser Modelländerung werden in Kapitel 3.4.2 sowohl für Simulationen mit stationären als auch für solche mit transienten Randbedingungen diskutiert.

### 3.4.2. Simulation stationärer und transienter Randbedingungen

Zur Bewertung der Effekte der Kosimulation werden zwei getrennte Untersuchungen durchgeführt: eine mit stationären Randbedingungen und eine mit transienten Randbedingungen. Typischerweise erfolgt die Auslegung des Wärmemanagements von Fahrzeugen an Hand ausgewählter quasistationärer Betriebspunkte [Betz u. a. 2005]. Hierbei werden neben den prozentual am häufigsten auftretenden Bedingungen vor allem Extrembedingungen betrachtet, da das System – z.B. der Kühlkreislauf – auch unter diesen Bedingungen funktionieren muss [Sargent 1988, Hager u. a. 2002]. Die Simulationen mit transienten Randbedingungen bilden zusätzlich die Dynamik des untersuchten Systems ab und ermöglichen so eine realistische Bewertung des Systems.

Für einen typischen stationären Betriebspunkt zeigt Abbildung 3.21 den grafischen Vergleich der Messdaten mit den Simulationsergebnissen des geschlossenen Modells und der Kosimulation. Zu erkennen ist eine stationäre Abweichung des Hochdrucks von ca. 7% zwischen den Simulationen und den Messdaten. Diese Abweichung lässt sich mit Schwierigkeiten und Ungenauigkeiten beim Messen der Luftgeschwindigkeiten an Verdampfer und Verflüssiger erklären. Im Diagramm ist ebenfalls erkennbar, dass durch die Kopplung keine signifikante Abweichung zum geschlossenen Modell eingeführt wird. Die einzelnen Komponentenmodelle wurden wie vorangegangen beschrieben validiert, nicht jedoch das zusammengesetzte Gesamtmodell. Die grafische Analyse gibt den Hinweis, dass auch das Gesamtsystem ohne weitere Anpassungen validiert ist.

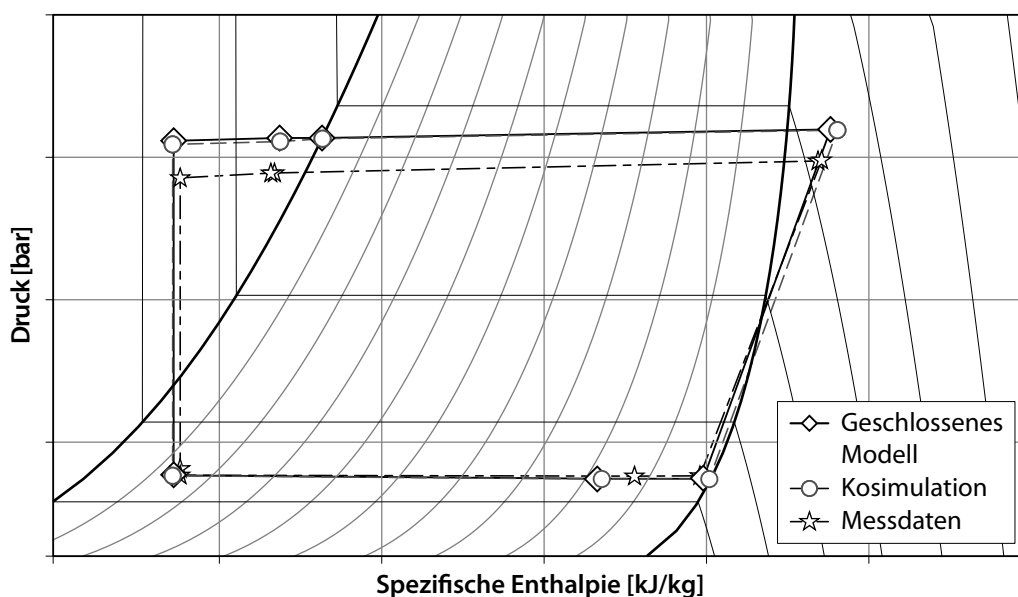


Abbildung 3.21.: Darstellung von Messdaten und Simulationsergebnissen im Druck-Enthalpie-Diagramm.

Da Messungen für die Systembewertung von Klimaanlage üblicherweise nur für stationäre Randbedingungen durchgeführt werden, können ausschließlich verschiedene Simulationsergebnisse mit transienten Randbedingungen verglichen werden. Um den Einfluss der Änderungen am Modell des internen Wärmeübertragers (vgl. Kapitel 3.4.1) und der Kopplung zu untersuchen, werden drei Modelle betrachtet:

#### 1. Geschlossen (ohne RC)

Modell bestehend aus zwei koaxial angeordneten, von Kältemittel durchströmten Rohren mit einer dünnen Trennwand.

#### 2. Geschlossen mit RC

Modell „Geschlossen“ mit zusätzlichem RC-Glied (Kombination von Wärmeleitwiderstand und -kapazität). Hierbei werden die Widerstände jeweils auf ca. 0,7% des Wärmeleitwiderstands, die Kapazität auf 70% der Kapazität der Wände eingestellt, wodurch eine zusätzliche Zeitkonstante von 0,1 s eingeführt wird.

### 3. Kosimulation (mit RC)

Aufgetrenntes System nach Abbildung 3.20. Die Parametrierung des Modells ist identisch mit dem Modell „Geschlossen mit RC“, die Synchronisationsrate wird auf konstant 0,1 s (die maximale Zeitkonstante im System) festgelegt.

Die Randbedingungen, mit denen diese drei Modelle beaufschlagt werden, sind Daten entnommen, die bei Messungen an einem Mittelklasse-PKW mit Klimaanlage aufgenommen wurden. Hierbei wurde als Fahrzyklus der NEFZ [Rat der EG 1970] verwendet. Es werden neben dem Kältemittelmassenstrom auch Hoch- und Niederdruck variiert (vgl. Abbildung 3.22). Für alle nachfolgenden Abbildungen wird das Zeitfenster von 500 s bis 600 s gewählt, in dem die Verläufe der Randbedingungen eine hohe Dynamik aufweisen.

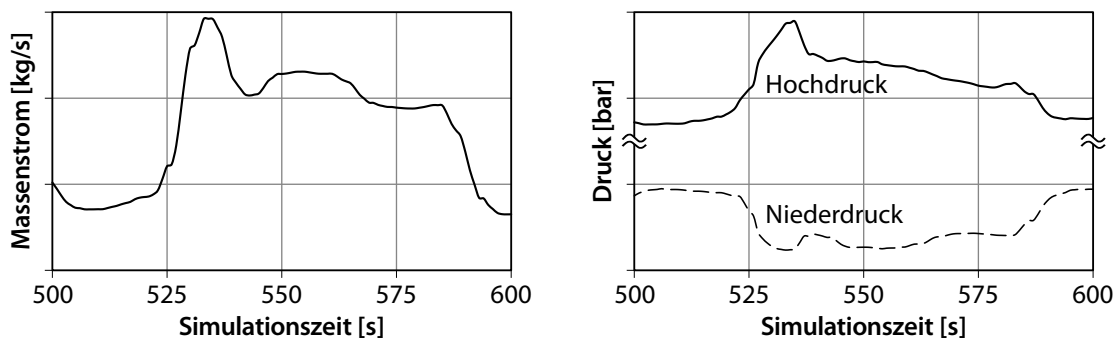


Abbildung 3.22.: Massenstromverlauf (links) und Druckverläufe (rechts) während einer NEFZ-Messung (Ausschnitt).

Die Wärmeströme im internen Wärmeübertrager auf der Hoch- und der Niederdruckseite für die drei Modelle sind in Abbildung 3.23 dargestellt. Das Einfügen der thermischen Kapazität führt zu einer Verringerung der Dynamik des Modells, aufgrund der zusätzlichen kleinen Wärmeleitwiderstände ergibt sich auf beiden Druckniveaus eine geringe stationäre Abweichung der simulierten Wärmeströme. Durch die Abstimmung der Synchronisationsrate der Kosimulation auf die Zeitkonstante des Wärmeübergangs im internen Wärmeübertrager liegen die Kurven des Modells „Geschlossen mit RC“ direkt über denen der Kosimulation.

#### 3.4.3. Auswirkungen auf Entwicklungs- und Rechenzeit

Die Verringerung der Entwicklungszeit neuer Produkte durch den Einsatz von Computersimulationen ist in der Literatur viel beschrieben (siehe z.B. Hackenberg [2006]). Durch Kosimulation kann diese Zeit weiter reduziert werden. Gründe hierfür sind z.B., dass die Teilmodelle nicht nur wie oben beschrieben einzeln entwickelt, getestet und validiert, sondern auch im Rahmen einer Gesamtsystems simulation als einzelne Teilsysteme simuliert werden können, wodurch sich deutlich einfachere Gleichungssysteme ergeben. Die direkte Auswirkung hiervon auf die Simulation von Thermofluidsystemen ist, dass der Aufwand des Anwenders, zu einem

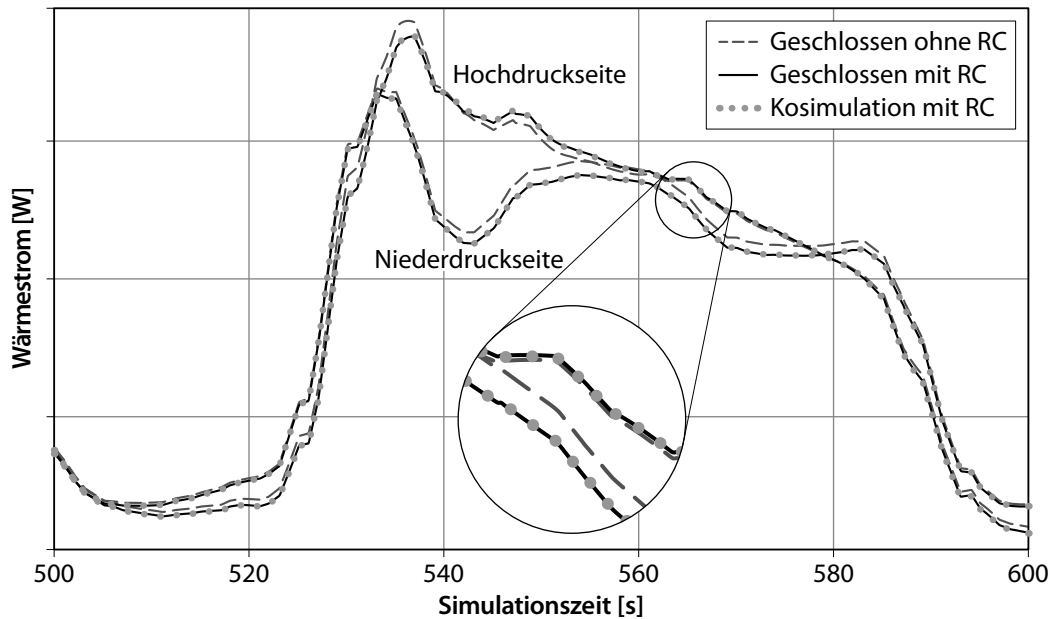


Abbildung 3.23.: Vergleich der simulierten Wärmeströme auf Hoch- und Niederdruckseite des internen Wärmeübertragers.

stabil und schnell initialisierenden System zu gelangen, für ein geschlossenes Gesamtsystem deutlich höher ist als bei einer Kosimulation – während die Kosimulation ohne Anpassungen schnell und stabil initialisiert, muss zunächst einige Zeit investiert werden, damit das geschlossene Modell überhaupt initialisiert. Je mehr Zeit investiert wird, desto stabiler und schneller kann die Initialisierung werden, die Zeit der gekoppelten Modelle aber nie ganz erreichen.

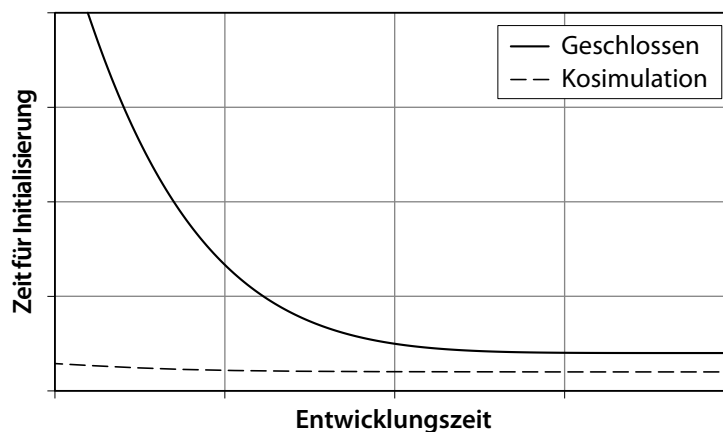


Abbildung 3.24.: Zeitbedarf der Initialisierung von geschlossenen und gekoppelten Systemen in Relation zur aufgewendeten Entwicklungszeit.

Ist eine stabile Initialisierung erreicht, rückt die Rechenzeit in den Vordergrund. Wie Abbildung 3.25 für stationäre Randbedingungen zeigt, weist die Kosimulation zu Beginn einen Geschwindigkeitsvorteil auf, der jedoch später von der geschlossenen Simulation wieder ausgeglichen wird. Hat die Simulation einen stationären Zustand erreicht, kann der Gleichungslöser seine interne Schrittweite und somit die Simulationsgeschwindigkeit erhöhen. Da für die

Kosimulation ein konstantes Synchronisationsintervall verwendet wird, ergibt sich der dargestellte lineare Verlauf. Zu beachten ist hierbei allerdings, dass das geschlossene Modell eine angepasste Parametrierung verwendet, damit das geschlossene System überhaupt initialisiert werden kann: Zum Zeitpunkt  $t = 0 \text{ s}$  sind sämtliche Transportbeziehungen für Wärmeübergänge und Druckabfälle auf konstante Werte gesetzt, zu späteren Zeitpunkten werden diese konstanten Werte auf physikalisch motivierte Korrelationen umgeschaltet, was sich auch in der Rechenzeit niederschlägt (in Abbildung 3.25 z.B. bei  $t \approx 37 \text{ s}$  und  $t \approx 43 \text{ s}$ ). Wie in Abbildung 3.24 dargestellt, kann mit weiterem Aufwand eine Verbesserung der Initialisierung und damit der Rechenzeit erreicht werden, auch eine Verwendung der komplexen Korrelationen von Beginn an ist so ggf. möglich. Die Kosimulation kann hingegen von Anfang an mit den korrekten Korrelationen rechnen. Ebenfalls in Abbildung 3.25 erkennbar ist der große Einfluss, den der Gleichungslöser auf die Simulationszeit hat. Während sich das geschlossene System ausschließlich mit dem Dassl-Löser [Petzold 1982] verwenden lässt, konnten für die verschiedenen Teilsysteme der Kosimulation auch unterschiedliche Löser eingesetzt werden. Wie der beispielhafte Vergleich von Radau- [Hairer u. Wanner 1996] und Dassl-Löser zeigt, gibt es hier erhebliches Potenzial zur Verringerung der Simulationszeit.

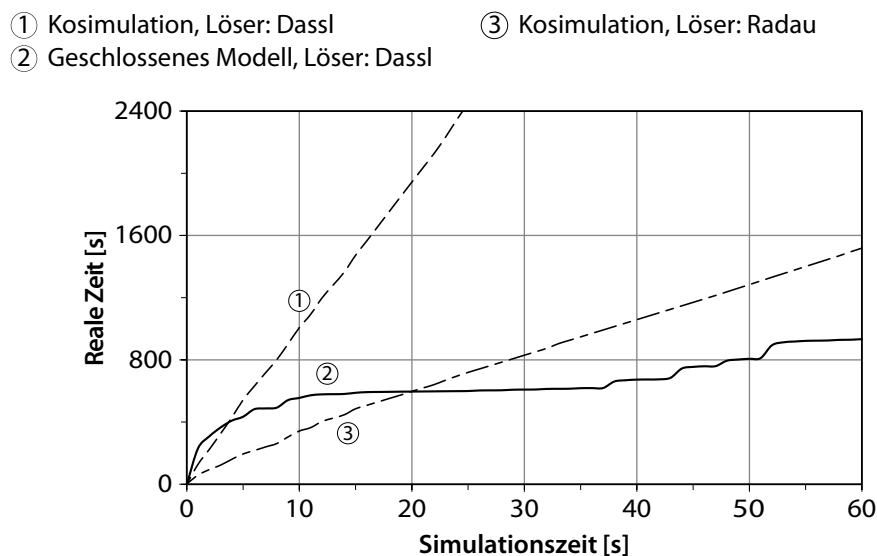


Abbildung 3.25.: Einfluss von Modellsetup und Gleichungslöser auf die Rechenzeit des Gesamtsystems nach Abbildung 3.15 und 3.19 bei stationären Randbedingungen.

Bei transienten Randbedingungen kann der Gleichungslöser die interne Schrittweite auch im geschlossenen System nicht so stark erhöhen, wie bei der Berechnung des stationären Grenzfalles (Abbildung 3.25). Hierdurch verringert sich der Vorteil der geschlossenen Simulation gegenüber der Kosimulation. Abbildung 3.26 zeigt die Verläufe der Rechenzeiten der drei Modelle des internen Wärmeübertragers (vgl. Kapitel 3.4.2) unter den in Abbildung 3.22 dargestellten Randbedingungen. Für die Kosimulation wurde als Löser Radau, für die geschlossenen Systeme Dassl verwendet.

- ① Kosimulation mit RC-Glied
- ② Geschlossenes Modell mit RC-Glied
- ③ Geschlossenes Modell ohne RC-Glied

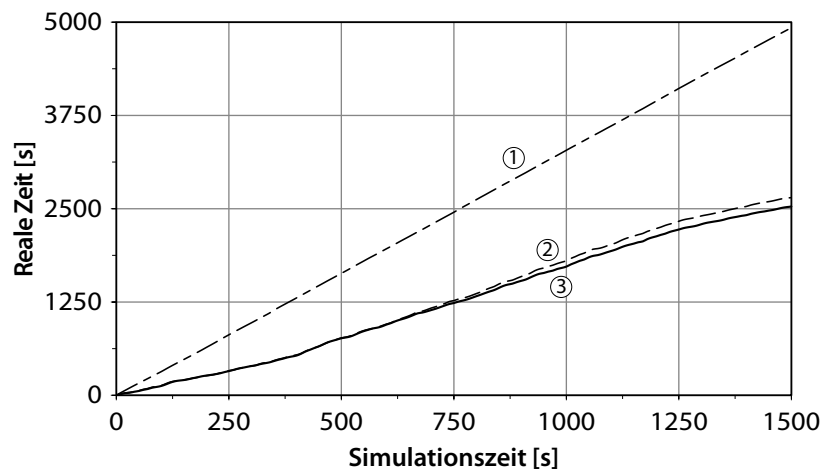


Abbildung 3.26.: Einfluss des Modellsetups für den internen Wärmeübertrager auf die benötigte Rechenzeit bei transienten Randbedingungen.

## 4. Umsetzung der Konzepte

Für die im Rahmen dieser Arbeit entwickelte Kosimulationsumgebung werden folgende Mindestanforderungen definiert:

- ♦ **Verteilte Simulation**

Die Simulationsprogramme sollen über das Netzwerk verteilt auf verschiedenen Computern ausgeführt werden.

- ♦ **Plattformunabhängigkeit**

Als Betriebssysteme sollen mindestens Microsoft Windows und Linux unterstützt werden.

- ♦ **Black Box**

Es muss davon ausgegangen werden, dass die interne Funktionsweise der einzelnen Teilsystemsimulationen nicht bekannt ist.

- ♦ **Einfache Integration von Simulationsprogrammen**

Die Anbindung der verschiedenen Simulationsprogramme soll möglichst einfach von staten gehen. Hierfür ist neben einem einfachen Interface vor allem die Unterstützung möglichst vieler Programmiersprachen und -techniken notwendig. Es ist davon auszugehen, dass die Programm-Schnittstellen der meist kommerziellen Simulationswerkzeuge nicht verändert oder erweitert werden können.

- ♦ **Datenaufbereitung und -auswertung**

Da eine Kosimulation aus der verteilten Berechnung mehrerer Modelle besteht, liegen die Simulationsergebnisse – die Werteverläufe der modellinternen Variablen – nicht zentral vor. Lediglich die während der Kosimulation ausgetauschten Werte stehen während und nach der Simulation zur Auswertung zur Verfügung. Für diese Auswertung sind Möglichkeiten der Aufbereitung – insbesondere der grafischen – aufzuzeigen.

Dieses Kapitel stellt die Funktionsweise und Umsetzung der im Rahmen dieser Arbeit entwickelten Kosimulationsumgebung vor. Zunächst wird aufbauend auf Kapitel 2 die Kommunikation zwischen den für die Kosimulation genutzten Programmen dargestellt. Ausgehend hiervon wird ein Arbeitsprozess von der Definition des zu untersuchenden Systems bis zur Durchführung einer Kosimulation präsentiert, der als Minimalprozess zu verstehen und als Grundlage für reale Prozesse in Unternehmen heranzuziehen ist. Abschließend werden Programme vorgestellt, die im Rahmen dieser Arbeit zur Auswertung der übertragenen Variablenwerte erstellt wurden.

Die entwickelte Kosimulationsumgebung ist in zwei Ebenen unterteilt: die Simulations- und die Kontrollebene. Hierbei dient die Simulationsebene dem Datenaustausch zwischen den einzelnen Simulationsprogrammen während einer laufenden Kosimulation, die Kontrollebene übernimmt die Ansteuerung der einzelnen Programme.

### 4.1. Kommunikation während der Simulation

Als Kommunikationsstruktur der Simulationsebene wird der in Kapitel 2.3 vorgestellte zentrale Server verwendet. Die an der Kosimulation beteiligten Programme kommunizieren somit nicht direkt miteinander, sondern ausschließlich mit dem Server. Dieser überträgt nicht nur die Variablenwerte zwischen den Simulationsprogrammen, sondern übernimmt auch die zeitliche Synchronisation der Clients, wodurch die notwendige Logik auf der Clientseite minimal gehalten wird. Zudem können einzelne Modelle leicht durch andere ausgetauscht werden (z.B. ein 1-D-Modell einer Fahrgastzelle durch ein 3-D-CFD-Modell), solange beide Modelle dieselben Schnittstellen – die ausgetauschten Variablen – bedienen.

Die Anbindung der Clients an die einzelnen Simulationsprogramme wird über verschiedene Techniken realisiert: Das direkte Einbinden von Programmcode in den Sprachen C, C++, Fortran und Python (z.B. über benutzerdefinierte Funktionen bei CFD-Programmen) ist ebenso erprobt wie das Aufrufen von Programmschnittstellen über COM (z.B. Flowmaster, Kuli) und das Einbinden spezieller Blöcke (z.B. Modelica, Simulink). Alle Clients basieren auf demselben Programmkern. Hierdurch werden neue Versionen des Programmkerns (nach Beheben von Fehlern, Implementieren neuer Funktionalitäten, ...) automatisch in jede Schnittstelle übernommen, wodurch die Qualität und Konsistenz der einzelnen Schnittstellen sichergestellt ist. Abbildung 4.1 zeigt ein Beispiel für ein Kosimulationssetup mit der Simulationsebene.

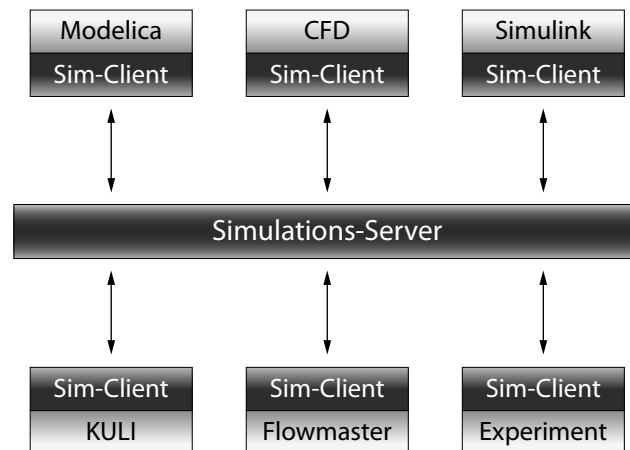


Abbildung 4.1.: Kommunikation in der Simulationsebene mit Clientbeispielen.



Die Simulationsebene beinhaltet sämtliche Programme und Programmschnittstellen, um eine Kosimulation durchführen zu können. Der Simulations-Server ist als eigenständiges Programm verfügbar und wird als erstes gestartet. Anschließend können die einzelnen Modellberechnungen durch die Simulationsprogramme gestartet werden, die wiederum über die integrierten Simulations-Clients mit dem Server kommunizieren. Die Konfiguration der Modelle erfolgt durch den Anwender in den entsprechenden Simulationsprogrammen. Während einer Kosimulation können auf Seite der Clients nach Puntigam u.a. [2006b] die drei Ausführungsphasen „Initialisierung“, „Durchführung“ und „Finalisierung“ unterschieden werden. Die Kommunikation zwischen den einzelnen Programmteilen innerhalb dieser Phasen zeigt Abbildung 4.2.

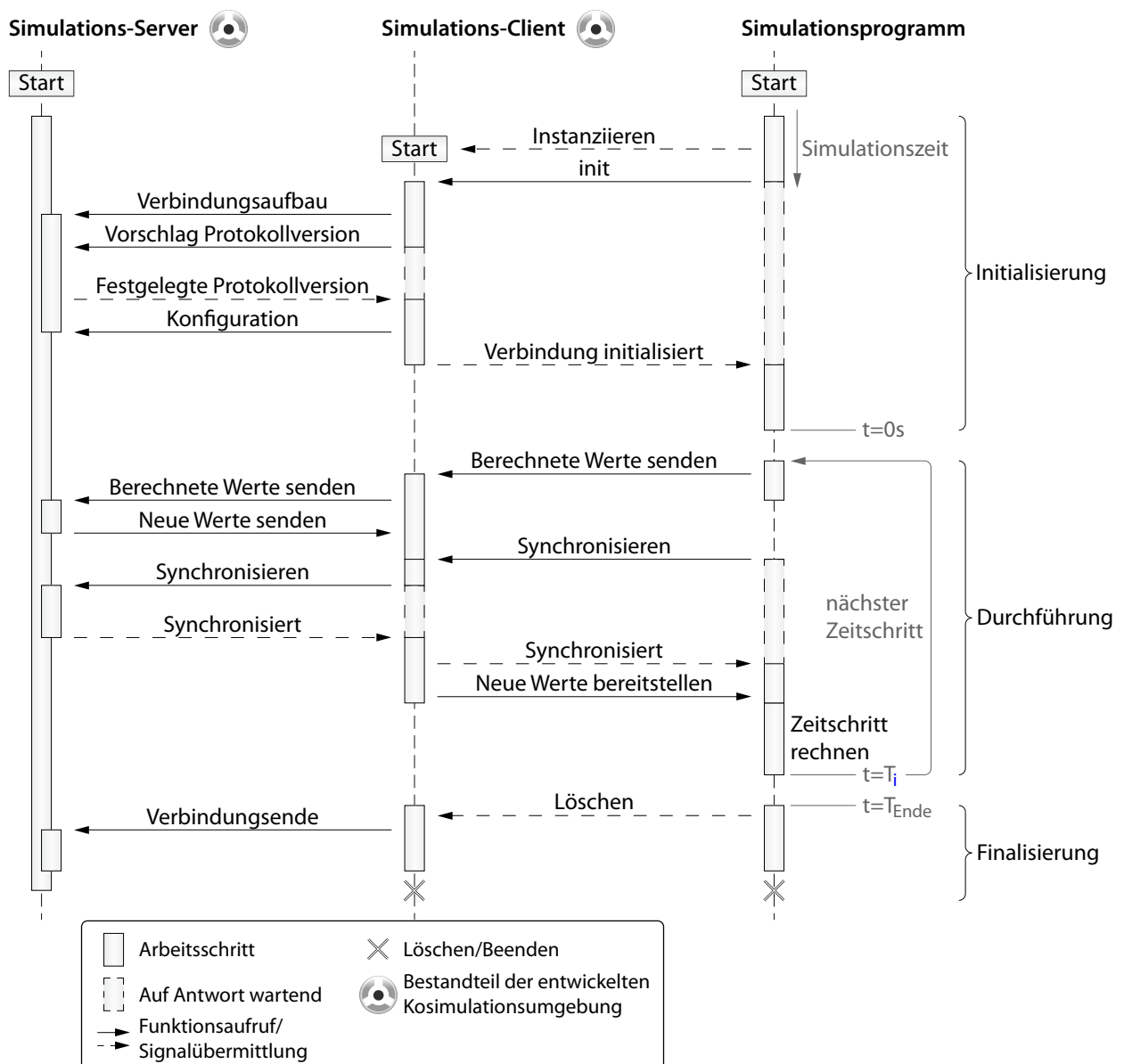


Abbildung 4.2.: Darstellung der Kommunikation zwischen Simulationsprogramm, Simulations-Client und Simulations-Server während einer Kosimulation im Sequenzdiagramm.

Der Inhalt der einzelnen Phasen ist somit im Einzelnen:

### 1. Initialisierung

Während der Initialisierungsphase des Modells wird auch die Verbindung zum Simulations-Server aufgebaut. Sobald die Verbindung steht, wird die verwendete Protokollversion bestimmt und die Konfiguration übertragen. Diese beinhaltet die Variablen, die vom Modell gesendet werden und empfangen werden sollen, sowie Übertragungseinstellungen.

### 2. Durchführung

In dieser Phase werden nach jedem berechneten Zeitschritt die Schritte (a) Variablenwerte senden, (b) Synchronisieren und (c) Variablenwerte empfangen durchgeführt. Hierbei kann auch die Länge des jeweils nächsten Rechenschritts festgelegt werden. Direkt nach der Initialisierung – zum Simulationszeitpunkt  $t = 0$  – wird die erste Synchronisation vorgenommen, um die errechneten Startwerte an die übrigen Modelle zu senden. Hierdurch wird ein konsistenter Datenstand erreicht.

### 3. Finalisierung

In diesem Schritt wird sämtlicher von der Verbindung zum Simulations-Server verwendeter Arbeitsspeicher wieder freigegeben. Auch wenn dies für die Kosimulation nicht relevant scheint, ist der Schritt gerade bei Clients notwendig, bei denen die Simulation nicht in eigenen Prozessen ausgeführt wird, da ansonsten unvorhergesehenes Verhalten auftreten kann (z.B. bei einem erneuten Herstellen der Verbindung zum Server, für die immer noch die alten Einstellungen gelten würden).

In der entwickelten Kosimulationsumgebung verwendet der Simulations-Server für den zeitlichen Abgleich der Clients einen auf der parallelen Synchronisation (siehe Kapitel 2.2.1) basierenden Algorithmus, der in Kapitel 3.3 vorgestellt ist (vgl. Abbildung 3.14). Für die Anbindung der einzelnen Berechnungsinstanzen einer parallelisierten CFD-Simulation stellt Correia [2009] einen Client vor, der die Daten der CFD-Instanzen gebündelt an den Simulations-Server weiterleitet.

## 4.2. Zentrale Ansteuerung der Simulationsprogramme

Werden viele Modelle zu einer Kosimulation zusammengeschaltet, ist es nicht praktikabel und zu zeitaufwändig, jedes von Hand zu konfigurieren und zu starten. Sind die Teilsimulationen auf mehrere Computer verteilt, wird dies weiter erschwert. Die Kontrollebene stellt die Infrastruktur zur Verfügung, die Kosimulation an zentraler Stelle zu konfigurieren, zu starten und zu stoppen. Hierzu wird auf jedem an der Kosimulation beteiligten Computer ein *Kontroll-Client* installiert, der dort als Hintergrundprozess ausgeführt wird. Integriert in den Kontroll-Client ist ein Netzwerkserver, zu dem sich ein zentraler *Kontroll-Server* verbinden kann. Dieser Kontroll-Server sendet Befehle, die Modelle zu konfigurieren, zu starten und ggf. zu stoppen. Nach dem Starten der einzelnen Modellberechnungen verbinden sich die dort integrierten TISC-Clients zum Simulations-Server, um die Kosimulation über die Simulationsebe-

ne durchzuführen. Um dem Anwender die Bedienung der Kosimulation weiter zu vereinfachen, sind Simulations- und Kontroll-Server im *TISC-Center* zusammengefasst. Abbildung 4.3 zeigt diese Struktur.

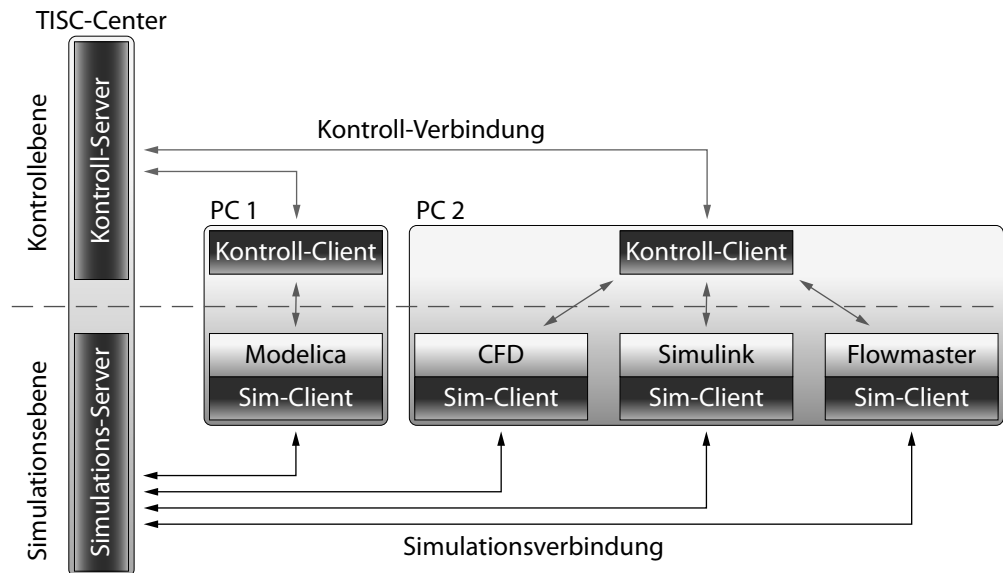


Abbildung 4.3.: Kommunikation in Simulations- und Kontrollebene mit TISC-Center und Clientbeispielen.

Das TISC-Center ist modular aufgebaut und stellt das zentrale Werkzeug der Kosimulation dar. Durch eine geeignete Aufbereitung der Informationen über die Modelle einschließlich Parametern und Variablen, benötigt der Anwender der Kosimulation kein Detailwissen über die Funktionsweise der Modelle, der verwendeten Simulationsprogramme oder der Kosimulationstechnik. Für die Konfiguration ist lediglich ein grundsätzliches Verständnis der Teilsysteme erforderlich. Die einzelnen im TISC-Center enthaltenen Module sind:

#### ♦ Konfiguration

Über dieses Modul werden die Modelle für die Kosimulation ausgewählt und deren Parameter eingestellt. Der Anwender wird hierbei durch Modell- und Variablenübersichten sowie der Erkennung von Konfigurationsproblemen und -fehlern unterstützt. Auch die Definition von Batchsimulationen ist möglich, um z.B. Parametervariationen durchzuführen.

#### ♦ Kontrolle

Mit den im Konfigurationsmodul definierten Einstellungen startet der Kontroll-Server, der über dieses Modul eingebunden ist, die einzelnen Simulationsmodelle auf den entsprechenden Client-Computern. Analog kann hierüber auch der Befehl zum Beenden der Simulation gegeben werden, wenn etwa bemerkt wird, dass die Ergebnisse den sinnvollen Bereich verlassen.

### ♦ **Simulation**

Der Simulations-Server ist als eigenes Modul im TISC-Center eingebunden. Durch die Integration in das TISC-Center kann über eine Verknüpfung mit Informationen des Konfigurationsmoduls der Funktionsumfang erweitert werden. So ist es möglich, das Anmelden und Verhalten der einzelnen Modelle zu überwachen und z.B. mit einer Fehlermeldung zu reagieren, wenn sich ein Modell nach einer bestimmten Zeit noch nicht am Server angemeldet hat. Ebenso kann die Kosimulation automatisch gestartet werden, sobald alle Modelle am Simulations-Server angemeldet sind, und automatisch gestoppt werden, sobald ein Modell die Verbindung zum Simulations-Server beendet hat – etwa weil die Berechnung abgebrochen ist.

### ♦ **Statusmeldungen**

Über die Kontrollebene können die Kontroll-Clients Informationen über den Client-Computer sowie den aktuellen Zustand der Simulationsmodelle und -programme schicken. Beispiele hierfür sind die aktuelle Prozessorauslastung des Client-Computers sowie Fehlermeldungen bei Programmabbruch.

### ♦ **Auswertung**

Verschiedene Module lassen sich unter dem Stichpunkt Auswertung zusammenfassen. Enthalten sind z.B. eine Tabelle mit den während der Simulation ausgetauschten Variablenwerten, die sich in einem anderen Modul auch grafisch darstellen lassen. Nach Simulationsende können automatisiert vorlagenbasierte Simulationsberichte generiert werden.

Um die Aufgabe der Konfiguration erfüllen zu können, benötigt das TISC-Center Zugriff auf Informationen über die zur Verfügung stehenden Modelle. Um diese Modellinformationen bereitstellen zu können, wurde im Rahmen dieser Arbeit das Programm *TIPS* (**TISC Interface and Parameter Setup**) entwickelt. Es untersucht automatisiert alle Modelle, die an der Kosimulation teilnehmen sollen, und extrahiert dabei die folgenden Informationen:

### ♦ **Simulationsparameter**

Konfiguration der Verbindung des Simulations-Clients zum Simulations-Server:

- IP-Adresse und TCP-Port des Simulations-Servers
- Synchronisationsmethode und -schrittweite
- Option, die aktuelle Simulationszeit als Variable zu senden
- Bezeichnung des Modells

### ♦ **Ausgetauschte Variablen**

Für jede Variable, die während der Kosimulation vom Modell gesendet oder empfangen wird, wird ein Startwert festgelegt, der über das TISC-Center nachträglich überschrieben werden kann. Zusätzlich können für jede Variable Einheit und Beschreibung angegeben werden, um die Konfiguration im TISC-Center zu erleichtern.

### ♦ **Modellparameter**

Der Modellierer kann Parameter in das Modell einfügen, deren Werte durch den Kosimulationsanwender im TISC-Center geändert werden können.

**♦ Simulationsinformationen**

Programmname und -version, mit der das Modell erstellt wurde, sowie Pfad und Name der Modelldatei.

Vom Modellierer ergänzt um folgende Angaben werden die Informationen in einer Datei mit der Dateiendung mip (**m**odel **i**nterface and **p**arameters) abgespeichert:

**♦ Nutzerparameter**

Zusätzlich zu den automatisiert ausgelesenen Modellparametern können weitere Parameter angelegt werden, um vor der Simulation Einstellungen verändern zu können. Ein Beispiel hierfür ist die gewünschte Kältemittelfüllmenge in der Klimaanlage eines PKW.

**♦ Modellinformationen**

- Name des Eintrags im TISC-Center
- Version
- Beschreibung

**♦ Modelliererinformationen**

Zur Sicherstellung der Nachverfolgbarkeit werden Name und Abteilung des Entwicklers abgespeichert, der für das Modell verantwortlich ist.

Die auf den Kontroll-Client-Computern abgelegten mip-Dateien können über das TISC-Center automatisiert eingelesen werden, woraufhin die Informationen wie oben beschrieben zum Konfigurieren der Kosimulation verwendet werden können.

### 4.3. Eingesetzte Übertragungstechnik

In der Literatur reichen die zur Übertragung der notwendigen Informationen eingesetzten Techniken vom einfachen Dateiaustausch (z.B. in Tonchev u. a. [2007]) bis zur hochkomplexen HLA (High Level Architecture). Einen Überblick über mögliche Techniken, zwischen einzelnen Prozessen Daten auszutauschen, geben z.B. Yahiaoui u. a. [2003], Dreier [2006] und Correia [2009]. In der Literatur sind unzählige Vergleiche von jeweils zwei oder drei Kopplungstechniken zu finden [z.B. Buss u. Jackson 1998, Schmidt u. a. 1995], die jedoch zu stark abweichenden Ergebnissen etwa hinsichtlich der Geschwindigkeit kommen, da dies stark von den Einsatzbedingungen wie Art und Anzahl der ausgetauschten Informationen abhängt.

Die Anforderungen an die Übertragungstechnik müssen somit abhängig von den Einsatzbedingungen definiert werden. Im vorliegenden Fall – der Kosimulation thermischer Systeme – sind diese:

**♦ Verteilte Simulation**

Standard-Netzwerktechnik soll verwendet werden, um die Kosimulation auf verschiedene Computer verteilt ausführen zu können.

- ♦ **Plattformunabhängigkeit**

Unterstützt werden sollen mindestens die Betriebssysteme Windows und Linux. Die Kommunikationspartner sollen auf unterschiedlichen Betriebssystemen ausgeführt werden können.

- ♦ **Kommunikationssicherheit**

Es muss sichergestellt sein, dass jede gesendete Nachricht beim Empfänger ankommt.

- ♦ **Geschwindigkeit**

Im Rahmen dieser Arbeit werden keine festen Anforderungen an die Geschwindigkeit der Übertragungstechnik gestellt. In Hinblick auf eine zukünftige Weiterentwicklung in Richtung einer echtzeitfähigen Kosimulation sind jedoch Techniken auszuschließen, die einen eindeutigen Geschwindigkeitsnachteil mit sich bringen.

Durch diese Anforderungen scheiden bereits die meisten Techniken aus. Viele sind entweder wie COM und Shared Memory auf einen Computer begrenzt oder wie Pipes nicht plattformübergreifend verfügbar. Auf Grund von Geschwindigkeitsvergleichen [z.B. Schmidt u. a. 1995, Hirano u. a. 1998] wird außerdem die String-basierte Kommunikation (z.B. über SOAP) ebenso ausgeschlossen wie das ausschließlich auf Java aufbauende RMI. Auch wenn HLA als Stand der Technik bei verteilter Simulation bezeichnet wird [Wilcox u. a. 2000, Straßburger u. a. 2003], wird es ausgeschlossen. Auf Grund der Ziele, für die es entwickelt wurde – die verteilte Simulation militärischer Anwendungen in Echtzeit mit tausenden Simulationsteilnehmern und mit menschlicher Interaktion –, bietet es sehr viele Möglichkeiten, die für die Kosimulation thermischer Systeme nicht benötigt werden. Da die Programmschnittstellen trotzdem alle Schnittstellen bedienen müssen, ergibt sich ein erheblicher Komplexitätsgrad, weshalb es nur sehr schwer einzusetzen ist [Taylor u. a. 2002].

Nach dem Ausschlussverfahren stehen das auf Sockets basierende CORBA (Common Object Request Broker Architecture) und eine eigene Implementierung einer Socket-Kommunikation zur Wahl. Da für CORBA zwar Schnittstellen zu diversen Programmiersprachen existieren, nicht jedoch z.B. zu Fortran, das im thermischen Bereich noch immer von vielen Simulationsprogrammen genutzt wird [Follen u. a. 2001], fällt die Entscheidung gegen CORBA. Darüber hinaus stellt CORBA aufgrund der Tatsache, dass es für die Kommunikation zwischen zwei beliebigen Anwendungen (nicht notwendiger Weise Simulationsprogrammen) entwickelt wurde, für die Simulation gleichzeitig zu viel und zu wenig Funktionalität zur Verfügung. Dies müsste durch eigene Erweiterungen oder spezielle Datenpakete gelöst werden [Trčka 2008].

Die Umsetzung erfolgt mit einer Eigenentwicklung auf Basis von TCP-Sockets. Über ein kompaktes Datenprotokoll werden Informationen über die aktuellen Variablenwerte, Simulationsfortschritt usw. zwischen den einzelnen Modellen und dem zentralen Server ausgetauscht. Sockets können mit verschiedenen Programmiersprachen erstellt und angesprochen werden, die Entscheidung fällt auf die Erstellung eines C++-Wrappers nach Schmidt [1992], wobei für die Implementierung die Klassenbibliothek Qt [Nokia 2010] verwendet wird. Hierdurch kann derselbe Quellcode ohne Anpassungen für verschiedene Plattformen verwendet werden.

## 5. Arbeitsprozess in Unternehmen

Der erste Schritt einer Kosimulation ist die Definition der Teilsystemgrenzen und der Variablen, die zwischen diesen Systemen ausgetauscht werden – die sog. Partitionierung. Wie bei jeder Modellierung gibt es auch bei der Kosimulation die Möglichkeiten, das Gesamtmodell nach dem Top-Down- oder dem Bottom-Up-Ansatz aufzubauen [Puntigam u.a. 2006b]. Hierbei beschreibt der Top-Down-Ansatz, dass ausgehend vom Gesamtsystem die Teilmodelle definiert und anschließend implementiert werden, während beim Bottom-Up-Ansatz aus bestehenden Teilmodellen ein Gesamtsystem zusammengesetzt wird. Wenzel u. a. [2007] beschreiben zusätzlich den Middle-Out-Ansatz, bei dem einige bestehende Modelle weiterverwendet werden, während das übrige System wie beim Top-Down-Ansatz aufgeteilt wird. In dieser Arbeit wird nicht näher auf die Partitionierung eingegangen, hierfür sei auf die Literatur (z.B. Lüdecke [2003], Kanth [2010]) verwiesen.

In der industriellen Praxis werden die Teilmodelle eines Gesamtsystems von den entsprechenden Fachabteilungen zur Verfügung gestellt [Lund u. a. 2008]. Dies hat den Vorteil, dass das in den Fachabteilungen vorhandene Fachwissen genutzt werden kann und sichergestellt ist, dass die Modelle dem aktuellen Stand der Entwicklung entsprechen. Zudem ist die Verantwortung für die Modelle und somit für ggf. auftretende Fehler geklärt.

Einen Minimalprozess vom Aufbau bis zur Durchführung einer Kosimulation, der als Ausgangspunkt für komplexere Prozesse zu verstehen ist, zeigt Abbildung 5.1. In dem Prozess werden drei Rollen unterschieden:

### 1. Kosimulationsverantwortlicher

Der Kosimulationsverantwortliche definiert das zu simulierende System und legt hierbei sowohl die Grenzen der Teilsysteme als auch die zwischen den Modellen auszutauschenden Variablen fest. Für jedes einzelne Modell wird so festgelegt, welche Komponenten es beinhaltet und welche Variablen gesendet und empfangen werden. Das Ergebnis dieses Schrittes ist eine abstrakte Systembeschreibung, die Ausgangspunkt aller weiteren Arbeitsschritte ist.

Zusätzlich vergleicht der Kosimulationsverantwortliche die von den einzelnen Modellentwicklern gelieferten Modelle mit der abstrakten Systembeschreibung und gibt die Modelle für die weitere Verwendung frei oder lässt sie nachbessern. Die freigegebenen Modelle stellt er über ein Netzlaufwerk, eine Datenbank oder eine ähnliche zentrale Datenablage den Anwendern zur Verfügung.

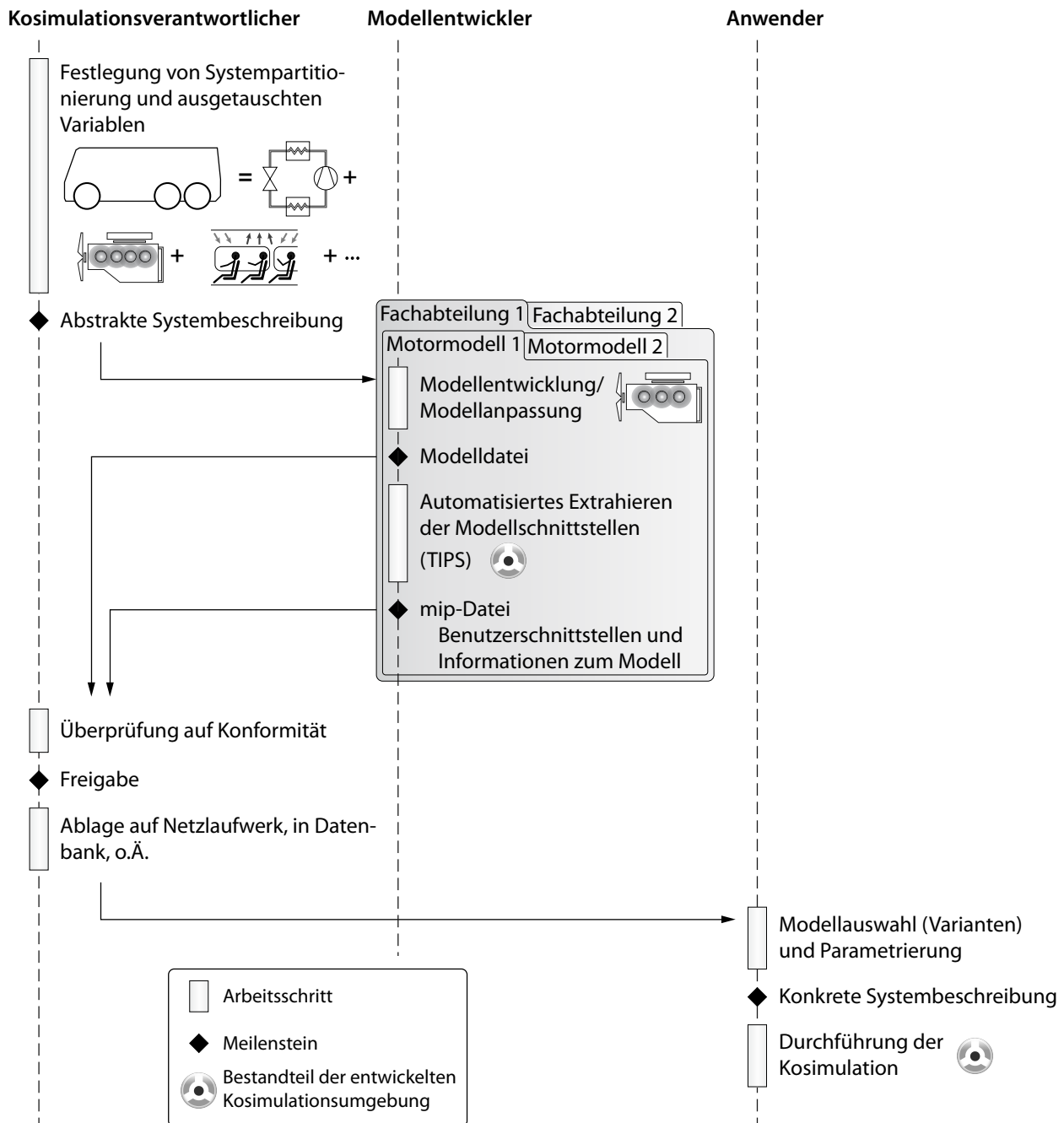


Abbildung 5.1.: Prozess zum Aufbau und Durchführen einer Kosimulation am Beispiel eines Reisebusses.

## 2. Modellentwickler

Modellentwickler gehören den Fachabteilungen an, in deren Verantwortung die einzelnen Teilsysteme fallen. Jeder Modellentwickler erhält über die abstrakte Systembeschreibung die Vorgaben, welche Informationen das Modell zu verarbeiten hat, und passt das im Allgemeinen bereits bestehende Modell dementsprechend an.

Mit Hilfe des Programms TIPS, das für die Kosimulationsumgebung im Rahmen dieser Arbeit entwickelt wurde, erstellt der Modellentwickler automatisch unterstützt eine



Informationsdatei, die das Modell beschreibt (ausgetauschte Variablen, vom Benutzer einstellbare Parameter, gültige Einsatzbereiche, ...) und zusätzlich Informationen über den Modellstand, die Fachabteilung und den Entwickler selbst enthält. Diese Datei liefert er zusammen mit dem Modell zurück an den Kosimulationsverantwortlichen zur Freigabe für die Kosimulation.

### 3. Kosimulationsanwender

Der Kosimulationsanwender bedient das TISC-Center (das zentrale Programm der im Rahmen dieser Arbeit entwickelten Kosimulationsumgebung) und führt mit diesem die Kosimulation durch. Basierend auf der abstrakten Systembeschreibung wählt er für jedes Teilsystem ein Modell aus den freigegeben Varianten aus, parametriert dieses und erstellt somit eine konkrete Systembeschreibung – das Gesamtmodell.

Abhängig vom Unternehmen, in dem die Kosimulation eingesetzt wird, sind Änderungen und Ergänzungen des Prozesses sinnvoll oder notwendig. Beispiele hierfür sind:

- ♦ **Teilsystemverantwortlicher**

Mit der Schaffung der Rolle eines Teilsystemverantwortlichen für jedes Modell wird eine Schnittstelle zwischen dem Kosimulationsverantwortlichen und den Modellentwicklern eingerichtet. Idealerweise gehört der Teilsystemverantwortliche der Abteilung an, die im Unternehmen für die Entwicklung des Teilsystems zuständig ist – der Abteilung des Modellentwicklers –, und weist somit mehr fachliche Kompetenz für das Teilsystem auf als der Kosimulationsverantwortliche. Hierdurch wird nicht nur der Kosimulationsverantwortliche entlastet, potentiell steigt zudem die Qualität der Teilsysteme an.

- ♦ **Modellmanagement**

Unabhängig vom Speicherort der Modelldateien (Netzlaufwerk, Datenbank, ...), gibt es zwei Möglichkeiten der Modellablage, wenn Teilsysteme in verschiedenen Gesamtsystemen eingesetzt werden können: (1) Erzeugung einer Kopie des Modells für jedes Gesamtsystem und (2) Verfeinerung der Ablagestruktur mit Einführung eines detaillierten Rechtemanagements zur Bestimmung, welche Modelle für welche Gesamtsysteme verwendet werden dürfen. Die erste Möglichkeit birgt durch den Mehraufwand bei der Pflege verschiedener Modellstände die Gefahr, dass die unterschiedlichen Gesamtsysteme mit inkonsistenten Modellständen simuliert werden. Die zweite Methode erzeugt zwar einen erhöhten Aufwand beim Aufbau der Kosimulation, es wird aber sichergestellt, dass konsistente Modellstände verwendet werden.

Ein Beispiel für ein mehrfach genutztes Teilsystem ist bei Fahrzeugherstellern der Motor. Typischerweise werden Fahrzeuge mit unterschiedlichen Motorkonfigurationen hergestellt, gleichzeitig können die verschiedenen Motoren in unterschiedliche Fahrzeugtypen eingebaut werden. Für die Kosimulation unterschiedlicher Gesamtsysteme – der Fahrzeugtypen – ist es sinnvoll, für identische Motoren dasselbe Modell zu verwenden, da hierdurch z.B. der Freigabeprozess nur einmal durchlaufen werden muss.



## 6. Simulation des thermischen Gesamtsystems eines Reisebusses

Dieses Kapitel stellt eine Anwendung der im Rahmen dieser Arbeit entwickelten und in den vorangegangenen Kapiteln vorgestellten Kosimulationsumgebung vor. Das Ziel dieser Kosimulation ist, das thermische Gesamtsystem eines Reisebusses abzubilden und unter Berücksichtigung aller Wechselwirkungen zwischen den einzelnen Teilsystemen untersuchen zu können.

Der Aufbau dieses Kapitels folgt dem in Kapitel 5 vorgestellten Arbeitsprozess. Die folgenden Unterkapitel sind dementsprechend:

### 6.1. Aufteilung in Teilsysteme

Definition, welche Teilsysteme für die Kosimulation des Gesamtsystems erstellt werden und welche Variablen zwischen diesen ausgetauscht werden.

### 6.2. Modelle der Teilsysteme

Beschreibung der für die einzelnen Teilsysteme erstellten Modelle.

### 6.3. Bestimmung der Kosimulationseinstellungen

Untersuchung, wie die Kosimulationseinstellungen – vor allem die Synchronisationsrate – zu wählen sind. Hierfür werden zwei Anwendungsfälle betrachtet, die Sprunganregungen für verschiedene Teilsysteme bedeuten.

### 6.4. Durchführung der Kosimulation: Reise Lissabon – Madrid

Simulation der Kühlung eines Reisebusses bei einer typischen Verwendung: Einer Reise von Lissabon (Portugal) nach Madrid (Spanien).

### 6.1. Aufteilung in Teilsysteme

Zur Untersuchung des thermischen Systems des Reisebusses werden die folgenden Hauptsysteme des Busses identifiziert:

#### 1. Kältekreislauf

Komponenten sind der über eine Kardanwelle mit dem Motor verbundene Kältemittelverdichter, das Aufdachklimagerät mit Verflüssiger, Sammler, Ventil und Verdampfer sowie ggf. zusätzliche Klimageräte im Bugschrank sowie im Fahrerruheraum, jeweils bestehend aus Ventil und Verdampfer.

#### 2. Kühl-/Heizkreislauf

Kombination aus Motorkühlkreislauf mit den Komponenten Motor, Hauptwasserkühler sowie Kühlmittelpumpe und dem Heizkreislauf bestehend aus Heizungswärmetauschern, die jeweils den Kältemittelverdampfern des Kältekreislaufs luftseitig nachgeordnet sind, und abhängig vom Lastfall Konvektorheizungen im Fußbereich des Innenraums.

#### 3. Innenraum

Fahrgastzelle mit Fenstern, Einbauten (Sitze, ...) und Passagieren.

#### 4. Längsdynamik

Mechanische Komponenten wie Reisebuskörper, Räder, Getriebe und Motor.

Da das Augenmerk der hier vorgestellten Simulation auf den thermischen Systemen liegt, wird die Längsdynamik vereinfacht abgebildet. Sie enthält zusätzlich ein Fahrermodell, das abhängig von der Sollgeschwindigkeit des Fahrzeugs Gas-, Brems- und Kupplungspedal sowie die Gangschaltung bedient.

Zur Festlegung von Fahr- und Umgebungszustand werden zusätzlich Systeme eingeführt, die nach Kapitel 3.2 als Randbedingungszeitreihen ausgeführt sind:

#### 5. Fahrprofil

Vorgabe von Route (Fahrtrichtung, Höhenprofil, Längen- und Breitengrad) sowie Fahrgeschwindigkeit.

#### 6. Klimatische Bedingungen

Berechnung des Umgebungszustands – Temperatur, Luftfeuchtigkeit, ... – für den aktuellen Standort und die aktuelle Uhrzeit. Um die Berücksichtigung der Solarstrahlung zu erlauben, werden die direkte und diffuse Solarstrahlung vorgegeben und zusätzlich Azimut- und Höhenwinkel der Sonne berechnet.

Abbildung 6.1 zeigt die Verschaltung der an der Kosimulation des Reisebusses beteiligten Modelle. Zu beachten ist, dass zur Erhöhung der Übersichtlichkeit nicht alle zwischen den Modellen ausgetauschten Variablen abgebildet sind. Eine vollständige Auflistung ist in Tabelle 6.1 zu finden.

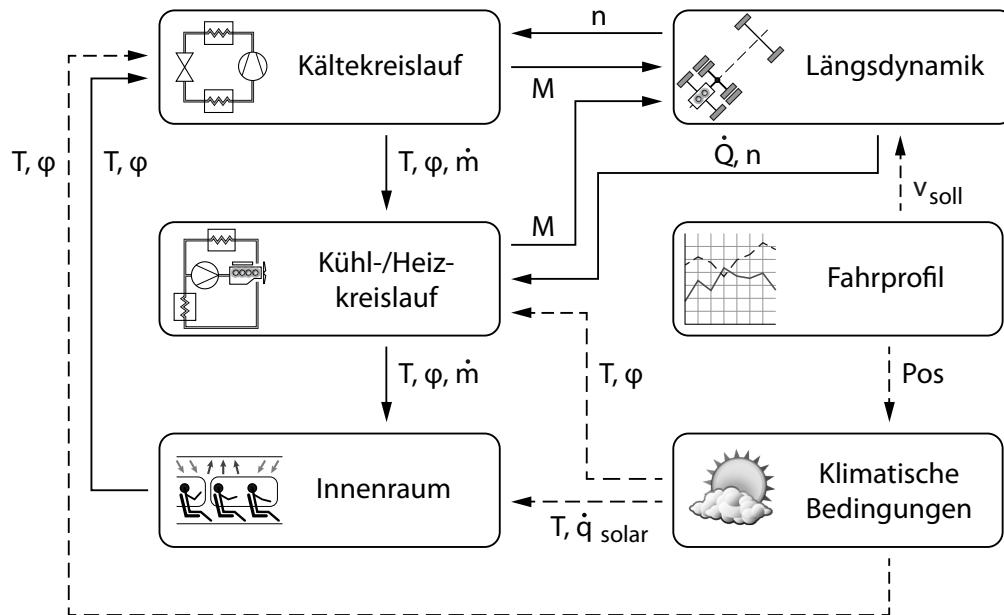


Abbildung 6.1.: Aufteilung des Gesamtsystems Reisebus in Teilsysteme mit Auswahl der ausgetauschten Variablen.

Die drei thermischen Teilsysteme Kältekreislauf, Kühl-/Heizkreislauf und Innenraum sind durch den Fluidpfad der Luft verbunden. Die aus dem Verdampfer des Kältekreislaufs austretende Luft wird durch den Heizungswärmetauscher des Kühl-/Heizkreislauks geleitet, bevor sie in den Innenraum eintritt. Die Informationen über den dort herrschenden Zustand werden zur Berücksichtigung der Umluft zurück an den Kältekreislauf übertragen, wo sich zusammen mit dem Frischluftanteil, dessen Zustand über die klimatischen Bedingungen bestimmt wird, der Eintrittszustand in den Verdampfer ergibt.

Mit einem im Längsdynamikmodell enthaltenen Fahrermodell wird aus dem aktuellen Fahrzeugzustand und der vom Fahrprofil vorgegebenen Sollgeschwindigkeit der aktuelle Getriebe-gang und damit die Motordrehzahl berechnet. Diese wird an den Kältekreislauf zur Bestimmung der Verdichterdrehzahl und den Kühl-/Heizkreislauf für die Kühlmittelpumpe und das Radiatorgebläse übertragen. Die für diese Komponenten benötigten Antriebsmomente werden als Lastmomente an die Längsdynamik übergeben.

Die Komponente Motor kann zwei Systemen zugeordnet werden: dem Kühl-/Heizkreislauf und der Längsdynamik. Wird eine Komponente mehrfach mit unterschiedlichen Modellen in verschiedenen Teilsystemen berechnet, kann es zu inkonsistenten Ergebnissen kommen. Aus diesem Grund wird für diese Arbeit festgelegt, dass der Motor der Längsdynamik zugeordnet wird. Deshalb wird die an das Kühlmedium übertragene Wärme im Längsdynamikmodell bestimmt und an den Kühl-/Heizkreislauf übergeben.

Teilsystem	Empfangene Variablen	Gesendete Variablen
Kältekreislauf	$n$ Motordrehzahl $T$ Außenlufttemperatur $p$ Außenluftdruck $\varphi$ Außenluftfeuchtigkeit $T$ Innenraumtemperatur $\varphi$ Innenraumluftfeuchtigkeit	$M$ Antriebsmoment Kältemittelverdichter $T$ Lufttemperatur am Verdampferaustritt $\varphi$ Luftfeuchtigkeit am Verdampferaustritt $\dot{m}$ Massenstrom am Verdampferaustritt
Kühl-/Heizkreislauf	$\dot{Q}$ Wärmeeintrag Motor $n$ Motordrehzahl $T$ Außenlufttemperatur $p$ Außenluftdruck $\varphi$ Außenluftfeuchtigkeit	$M$ Drehmomente Pumpe u. Ventilator $T$ Lufttemperatur nach Heizungswärmeübertrager $\varphi$ Luftfeuchtigkeit nach Heizungswärmeübertrager $\dot{m}$ Massenstrom nach Heizungswärmeübertrager $\dot{Q}$ Wärmeströme Heizungen
Innenraum	$T$ Außenlufttemperatur $\dot{q}$ Solarstrahlung $A$ Azimut der Sonne $h$ Höhenwinkel der Sonne $T$ Lufttemperatur nach Heizungswärmeübertrager $\varphi$ Luftfeuchtigkeit nach Heizungswärmeübertrager $\dot{m}$ Massenstrom nach Heizungswärmeübertrager $\dot{Q}$ Wärmeströme Heizungen	$T$ Innenraumtemperatur $\varphi$ Innenraumluftfeuchtigkeit $PMV$ Aktuelle thermische Behaglichkeit
Antriebsstrang	$v_{soll}$ Sollgeschwindigkeit $M$ Drehmomente Kältemittelverdichter, Pumpen u. Ventilatoren	$n$ Motordrehzahl $\dot{Q}$ Wärmeeintrag Motor
Fahrprofil		$v_{soll}$ Sollgeschwindigkeit $Lat$ Geografische Breite $Lon$ Geografische Länge $\alpha$ Fahrtrichtung $t$ Fahrzeit
Klimatische Bedingungen	$Lat$ Geografische Breite $Lon$ Geografische Länge $t$ Fahrzeit	$T$ Außenlufttemperatur $p$ Außenluftdruck $\varphi$ Außenluftfeuchtigkeit $\dot{q}$ Solarstrahlung $A$ Azimut der Sonne $h$ Höhenwinkel der Sonne

Tabelle 6.1.: Kopplungsvariablen der einzelnen Teilsysteme.

## 6.2. Modelle der Teilsysteme

### 6.2.1. Kältekreislauf

Der betrachtete Kältekreislauf besteht aus den Bauteilen Verdichter, Gaskühler, Sammler, Ventil und Verdampfer. Als Kältemittel wird R-744 ( $\text{CO}_2$ ) verwendet. Abbildung 6.2 zeigt das Verschaltungsschema des Kältekreislaufmodells. Als Ventil kommt ein elektronisches Expansionsventil (EXV) zum Einsatz, dessen Strömungsquerschnitt so geregelt wird, dass sich ein energieoptimaler Hochdruck einstellt. Der jeweilige Strömungsquerschnitt wurde in einer vorgelagerten Optimierungssimulation bestimmt. Ein weiterer Regler bestimmt den Umluftanteil an der vom Verdampfer abgekühlten Luft. Für den Aufbau des Kreislaufmodells werden Komponenten der Modelica-Bibliothek TIL [Gräber u. a. 2009] verwendet, einzelne Modelle wie der Kompressor und das EXV sind Ergänzungsbibliotheken entnommen.

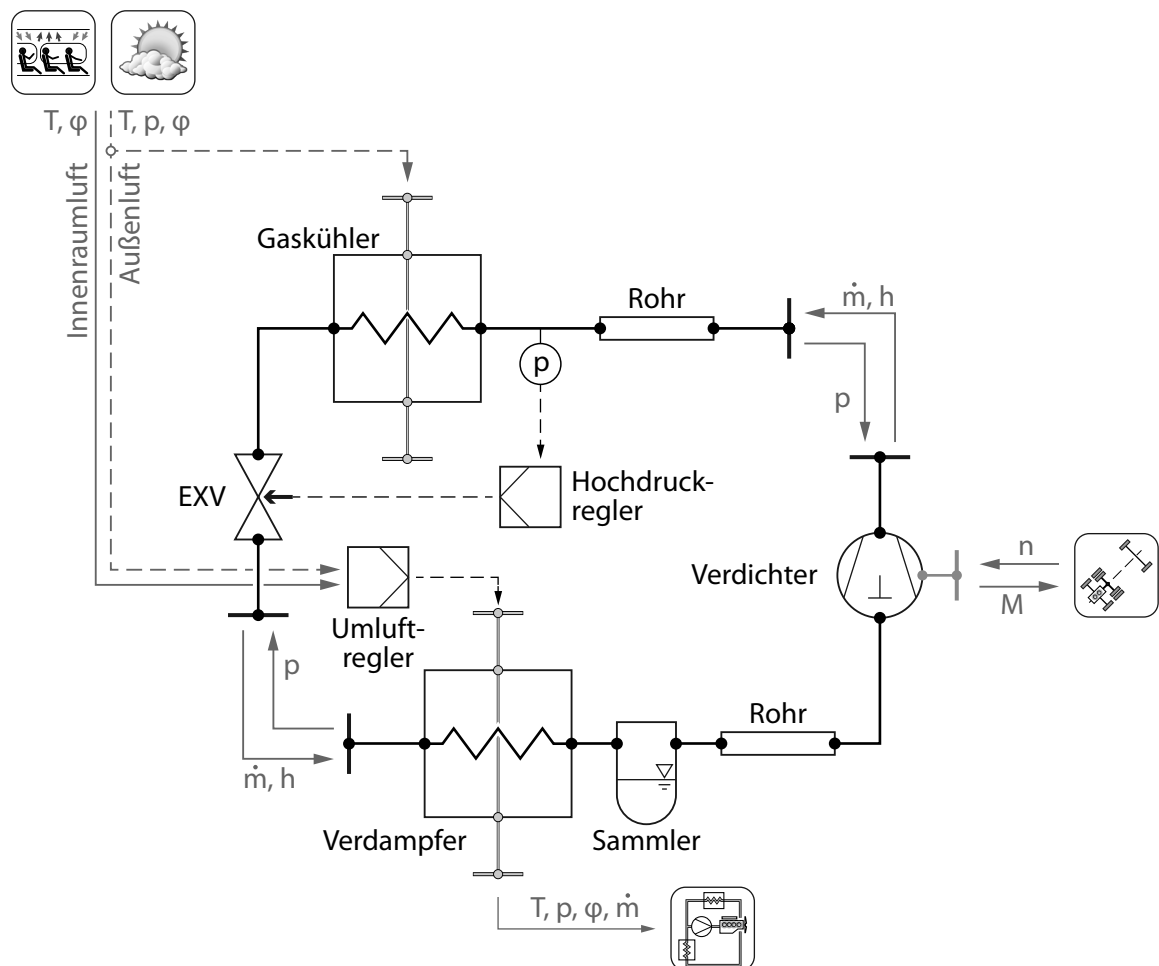


Abbildung 6.2.: Schema des Kältekreislaufmodells mit Auftrennungen des Fluidstroms nach Verdichter und elektronischem Expansionsventil (EXV) sowie Variablen, die zwischen den Teilsystemen und mit angrenzenden Systemen (vgl. Abbildung 6.1) ausgetauscht werden.

Der Kältemittelstrom wird in Strömungsrichtung nach dem Verdichter und nach dem Ventil aufgeschnitten und die so erzeugten Teilsysteme während der Kosimulation als unabhängige Teilsysteme einzeln berechnet. Wie in Kapitel 3.4 wird hierdurch das Gleichungssystem des Kältekreislaufmodells in zwei kleinere Gleichungssysteme geteilt, die insgesamt schneller berechnet werden können. Zwischen den zwei Teilsystemen werden, wie in Abbildung 6.2 gezeigt, die Variablen Druck, Enthalpie und Massenstrom ausgetauscht. In TIL wird für jedes Druckniveau ein zentralisierter Druckzustand eingeführt, um die Anzahl der differentiellen Zustände zu verringern [Lemke 2004]. Somit wird jedes Druckniveau durch eine differentielle Variable repräsentiert, die nach Kapitel 3.3 aus dem System gesendet werden muss. Dieser Empfehlung folgend werden die Drücke von den Druckniveaus zu der fluidfördernden Komponente (hier dem Verdichter) bzw. dem Drosselorgan (dem Ventil) übertragen. Die dem Druck nach Tabelle 3.1 zugeordnete Prozessgröße Volumenstrom wird hier mit der Dichte zum Massenstrom verknüpft und entgegen der Druckinformation übertragen. Zur Bestimmung des Kältemittelzustands wird die spezifische Enthalpie als zweite Zustandsgröße in Richtung der Massenstrominformation gesendet.

Als Eingangsgrößen von den übrigen Systemen werden die Luftzustände an Verflüssiger (Temperatur, Druck und Luftfeuchte der Außenluft) und Verdampfer (Temperatur, Druck und Luftfeuchte von Außenluft und Abluft des Innenraums) empfangen. Da der Verdichter über einen Riemenantrieb mit dem Motor verbunden ist, wird dessen Drehzahl als weitere Eingangsgröße empfangen. Als Regelmöglichkeit des Kältemittelmassenstroms kann eine Zylinderbank des Verdichters abgeschaltet werden, wodurch sich das genutzte Hubvolumen halbiert. Im für diese Arbeit genutzten Modell ist die Regelung so konfiguriert, dass der Verdichter oberhalb einer Außentemperatur von 35 °C mit vollem Hubvolumen arbeitet, unterhalb von 35 °C mit halbem. Neben dem benötigten Antriebsmoment des Verdichters wird der Luftzustand am Austritt des Verdampfers als Ausgangsgröße verschickt.

### 6.2.2. Kühl-/Heizkreislauf

Der Kühl-/Heizkreislauf erfüllt im betrachteten Reisebus die Funktionen der Motorkühlung und der Innenraumheizung. Abbildung 6.3 zeigt das für den Kreislauf verwendete Modellschema. Da im Anwendungsbeispiel dieser Arbeit ausschließlich ein Kühlfall betrachtet wird, sind in dem Schema die Komponenten vernachlässigt, die ausschließlich bei einem Heizfall zum Einsatz kommen (Konvektorheizungen im Fußraum, Kraftstoffbrenner, ...). Wie für das Kältekreislaufmodell wird auch für dieses Modell die Komponentenbibliothek TIL verwendet.

Die Hauptpumpe ist über einen Riemen mit der Kurbelwelle des Motors verbunden, wodurch sich eine direkte Abhängigkeit zwischen der Pumpen- und der Motordrehzahl ergibt. Als Ausgabe liefert das Modell das benötigte Antriebsmoment. Wie in Kapitel 6.1 ausgeführt, wird der Motor vollständig im Modell der Längsdynamik berechnet, weshalb hier lediglich der vom Motor in das Kühlmittel eingebrachte Wärmestrom empfangen wird. Die vom Motor eingebrachte Wärme wird über den Hauptwasserkühler abgeführt, dessen Durchströmung abhängig



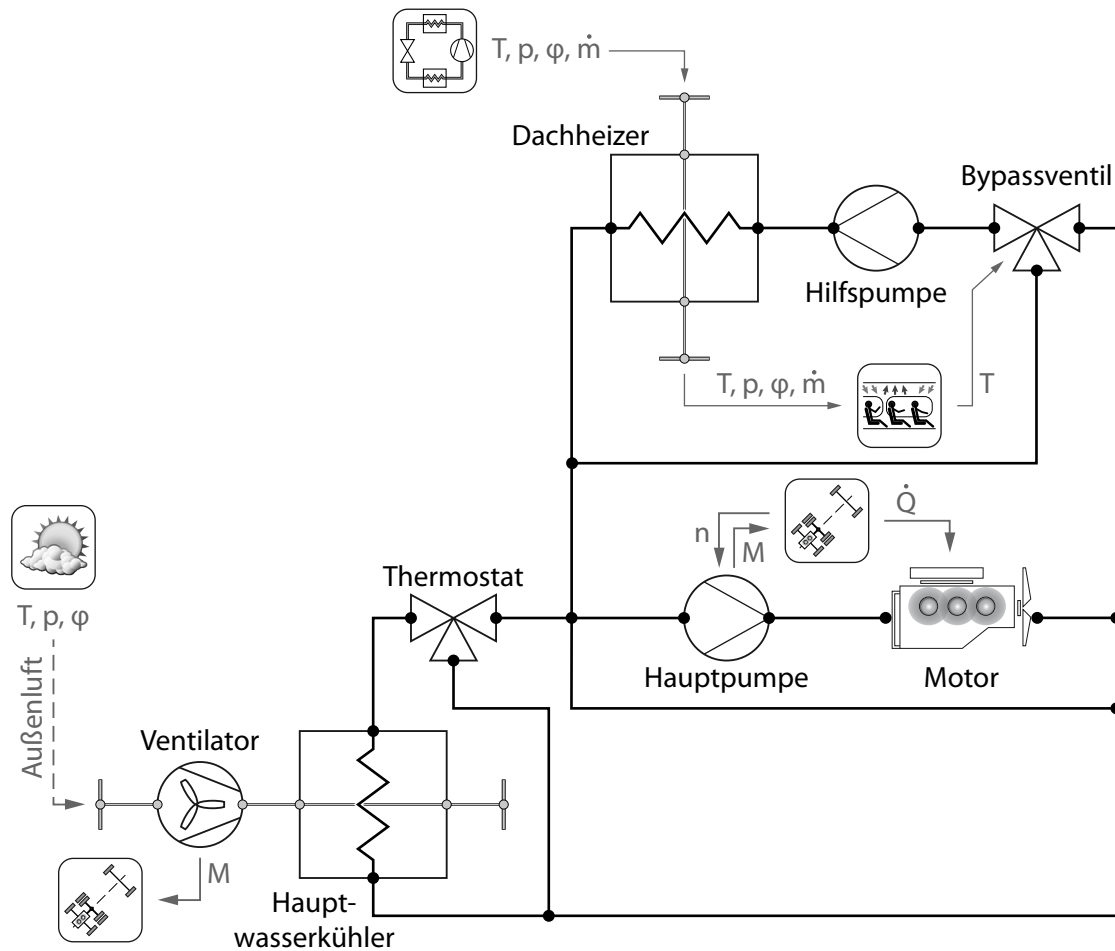


Abbildung 6.3.: Vereinfachtes Schema des Kühlkreislaufmodells ohne Rohre, mit gesendeten und empfangenen Variablen. Komponenten für Heizungsfunktion sind nicht dargestellt.

von der Kühlmitteltemperatur über ein Thermostat geregelt ist. Um den Höhenunterschied zum Dach zu ermöglichen, ist eine zusätzliche Pumpe notwendig. Der Dachheizer erwärmt die Luft, die aus dem Verdampfer des Kältekreislaufs austritt, bevor sie in den Fahrgastraum geleitet wird. Die Regelung dieser Aufheizung erfolgt über ein Bypassventil, das abhängig von der Innenraumtemperatur den Kühlmittelstrom durch den Dachheizer einstellt.

### 6.2.3. Innenraum

Der Innenraum wird mit der Bibliothek HumanComfort [Michaelsen u. Eiden 2009] modelliert, die auf die Modelica Standard Library in der Version 3.1 aufbaut. Ursprünglich dafür entwickelt, den thermischen Komfort in klimatisierten Räumen von Gebäuden zu berechnen, ermöglicht die HumanComfort-Bibliothek seit Einführung der Version 1.0 auch die Betrachtung mobiler Anwendungen.

Im Rahmen dieser Arbeit wurde ein Modell erstellt, das den Fahrgastraum in mehrere Sektionen unterteilt: (1) Windschutzscheibe, Fahrer und erste Sitzreihe, (2) Sitzreihen bis zur

Sektion 3, (3) Sitzreihen, über denen das Klimamodul (bestehend aus Verdampfer des Kältekreislaufs und Dachheizer des Kühl-/Heizkreislaufs) installiert ist, (4) restliche Sitzreihen und Heckscheibe (vgl. Abbildung 6.4).

Die Sektionen bestehen jeweils aus einem ideal durchmischten Kontrollvolumen sowie Seitenscheiben, Dach und Fußboden. Zur Belüftung wird Luft in Sektion 3 abgesaugt, in den übrigen Sektionen wird Luft eingeblasen. Der Umluftanteil der eingeblasenen Luft wird über einen Regler im Kältekreislaufmodell bestimmt (vgl. Abbildung 6.2). Alle Sektionen sind zudem mit Konvektorheizungen im Fußraum ausgestattet. Somit ergeben sich Wärmeströme von der Sonne über die Außenhaut (per Strahlung durch die Fenster und per Strahlung, Wärmeleitung und Konvektion durch die Blechteile) sowie von Klimamodul und Heizung in den Innenraum. Alle Kontrollvolumina sind mit Fahrgästen besetzt, die als zusätzliche Wärme- und Wasserquelle fungieren.

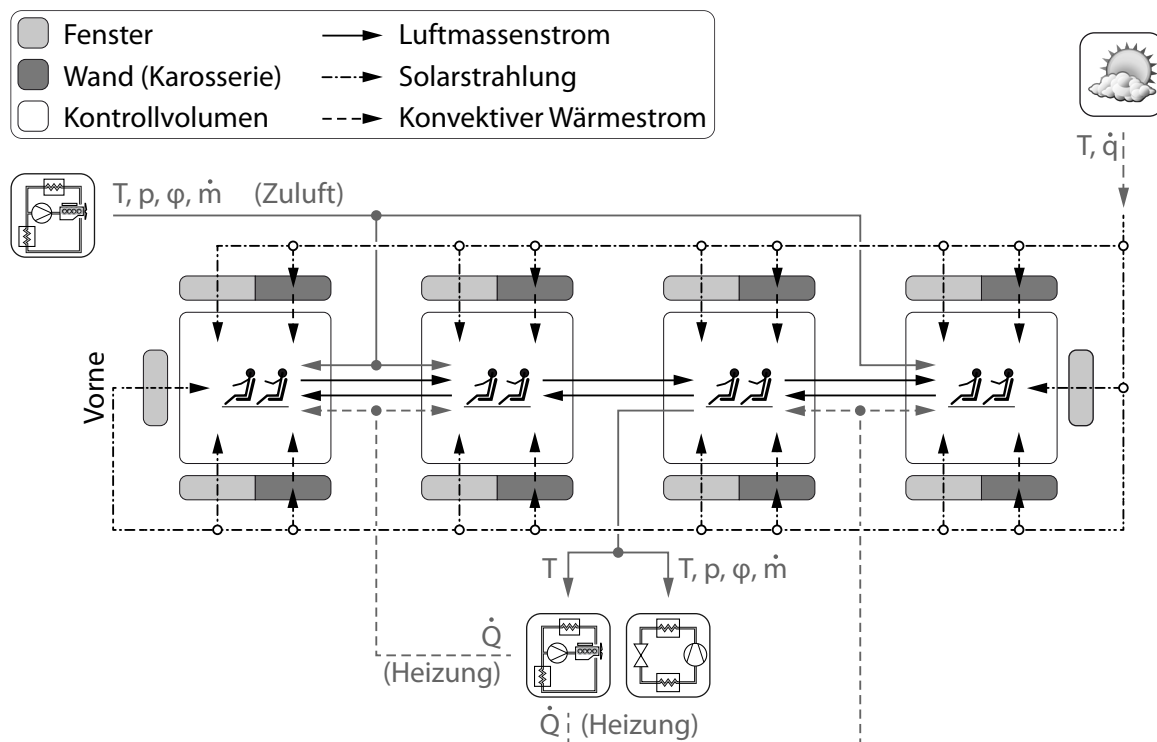


Abbildung 6.4.: Schema des Innenraummodells in Aufsicht: Aufteilung in vier Kontrollvolumina mit Fahrgästen, zugeordneten Wärme- und Luftströmen sowie mit angrenzenden Systemen ausgetauschten Variablen.

#### 6.2.4. Längsdynamik

Das Modell der Längsdynamik beinhaltet den Antriebsstrang des Reisebusses (Motor, Getriebe, Differenzial, ...), den Fahrzeugkörper (die Karosserie) sowie ein Fahrermodell. Für den aktuellen Fahrzustand – Geschwindigkeit, Lage, Beschleunigungen – werden Luft- und Fahrwiderstand sowie die Reibmomente und somit das vom Motor für den Vortrieb bereitzustellende

Antriebsmoment berechnet. Ergänzt um die notwendigen Antriebsmomente der Nebenaggregate (Kältemittelverdichter, Kühlmittelpumpe, Ventilator, ...) ergibt sich so die Drehmomentenanforderung an den Motor.

Neben der Antriebsleistung erzeugt der Motor Wärme, die über das Kühlwasser an alle angeschlossenen Systeme wie Kältekreislauf und Innenraum übertragen wird. Für die Güte der Simulationsergebnisse ist deshalb eine korrekte Beschreibung der vom Motor abgegebenen Wärmeströme essentiell [Betz u. a. 2005]. Das im Rahmen dieser Arbeit verwendete Motormodell basiert auf aus Messungen abgeleiteten Kennfeldern z.B. für die ins Kühlwasser abgegebene Leistung und den Kraftstoffverbrauch. Durch thermische Kapazitäten wird ein transientes Verhalten erzielt. Da bei Reisebussen die Aufheizphase einen verschwindend geringen Einfluss auf die gesamte Betriebszeit hat, reichen Modelle mit wenigen Masseknoten für die transiente Simulation aus [Hager u. a. 2003]. Für das Modell wird die Modellierungssprache Modelica in der Version 2.2.2 verwendet, die Modelle der einzelnen Komponenten wurden entweder neu erstellt oder der Modelica Standard Library entnommen.

Das Fahrermodell reagiert durch Bedienen von Gas-, Brems- und Kupplungspedal sowie der Gangschaltung auf die vom Fahrprofil empfangene Sollgeschwindigkeit. Ausgabegrößen an die angeschlossenen Teilsysteme sind die Motordrehzahl und der an das Kühlmittel abgeführte Wärmestrom.

### 6.3. Bestimmung der Kosimulationseinstellungen

Bevor Untersuchungen mit der Kosimulation durchgeführt werden können, müssen die zu simulierenden Modelle ausgewählt und parametrisiert werden. Die Modelle sind in Kapitel 6.2 beschrieben, deren Parametrierung neben Randbedingungen wie Start- und Initialwerten für verschiedene Modellvariablen vor allem die Kosimulationseinstellungen betrifft – hier vorrangig Synchronisationsrate sowie die Wahl und Einstellung von Glättung und Extrapolation. Um die Werte für die Parameter zu finden, werden zwei Testszenarien betrachtet:

#### 1. Ausrollen

Nach einer Konstantfahrt mit  $50\text{ km/h}$  wird das Kupplungspedal betätigt, so dass der Bus bei Leerlaufdrehzahl ausrollt. Hierdurch ergibt sich ein Sprung in der Motordrehzahl und damit auch in der Klimaverdichterdrehzahl, was sich direkt auf die Simulation des Kältekreislaufs und die Kühlung des Innenraums auswirkt.

#### 2. Tunnelfahrt

Simuliert wird die Einfahrt in einen Tunnel, also das Wegfallen der Solarstrahlung. Dies ermöglicht die Untersuchung des Verhaltens vor allem der Teilsysteme Innenraum und Kühl-/Heizkreislauf (wegen der dem Verdampfer nachgeschalteten Gegenheizung) bei sprunghafter Systemanregung.

Als Randbedingungen für beide Szenarien werden eine Umgebungstemperatur von  $35^{\circ}\text{C}$  und eine relative Luftfeuchte von 25% festgelegt. Die diffuse Solarstrahlung wird auf  $165\text{ W/m}^2$ , die direkte Solarstrahlung auf  $750\text{ W/m}^2$  gesetzt. Diese Bedingungen sind z.B. auf der in Kapitel 6.4 gezeigten Reise in Portugal und Spanien anzutreffen.

In beiden Szenarien werden zunächst  $1000\text{ s}$  konstante Bedingungen simuliert, damit sichergestellt ist, dass die Systeme einen stationären Zustand erreicht haben. Zum Zeitpunkt  $1000\text{ s}$  wird der Motor ausgekuppelt bzw. in den Tunnel eingefahren. Da der stationäre Zustand und das Systemverhalten nach Änderung der Randbedingungen von Interesse ist, werden in den folgenden Diagrammen zeitliche Ausschnitte dargestellt. Das erste Szenario zielt primär auf die Untersuchung des Kältekreislaufs ab, das zweite auf die des Innenraums.

### 6.3.1. Ausrollen

Abbildung 6.5 zeigt die Fahrzeuggeschwindigkeit für die oben beschriebenen Randbedingungen: Nach Vorgabe einer konstanten Fahrzeuggeschwindigkeit von  $50\text{ km/h}$  wird zum Zeitpunkt  $1000\text{ s}$  der Motor ausgekuppelt. Hierdurch fällt die Motor- und damit die Kältemittelverdichterdrehzahl sprunghaft ab, was sich direkt auf die Simulation des Kältekreislaufs auswirkt. Somit kann vor allem das Verhalten des Kältekreislaufs untersucht werden. Im Folgenden werden Kosimulationen mit verschiedenen Synchronisationsraten –  $0,1\text{ s}$ ,  $0,5\text{ s}$  und  $1,0\text{ s}$  – vorgestellt.

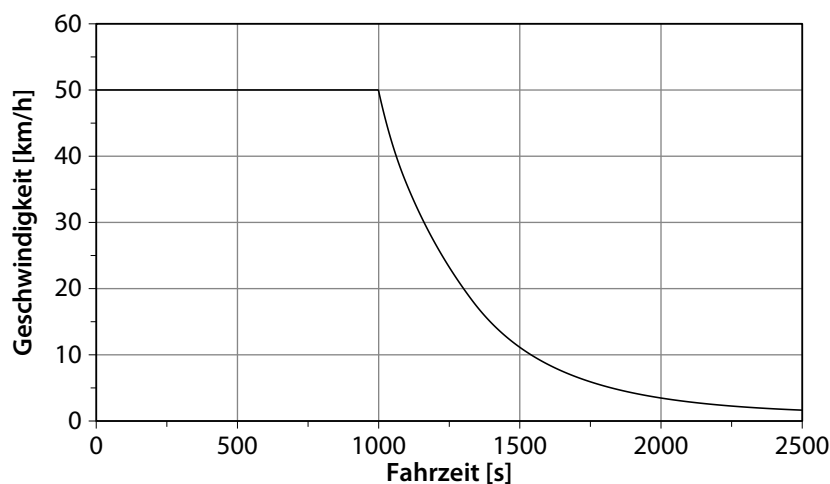


Abbildung 6.5.: Randbedingung Testszenario 1: Ausrollen des Reisebusses von  $50\text{ km/h}$  nach Auskuppeln des Motors bei  $1000\text{ s}$ .

Durch das Abfallen der Drehzahl des Kältemittelverdichters kommt es zu einem Einbruch der am Verdampfer übertragenen Kälteleistung, wie Abbildung 6.6 zeigt. Dass der von den Passagieren abgegebene Wärmestrom sinkt, ist dadurch zu erklären, dass die Kälteleistung nicht mehr ausreicht, um den Innenraum auf den gewünschten  $22^{\circ}\text{C}$  zu halten. Wie Abbildung 6.7 zu entnehmen ist, steigt die Innenraumtemperatur ab dem Auskuppeln des Motors

an, wodurch die Temperaturdifferenz zwischen Passagieren und Innenraum verkleinert wird und sich somit ein geringerer Wärmestrom einstellt. Mit dem Anstieg der Innenraumtemperatur ist auch die starke, sprungartige Änderung der Verdampferleistung bei 1025 s zu erklären, da sich zu diesem Zeitpunkt durch die verwendete Steuerung die Umlufterate ändert.

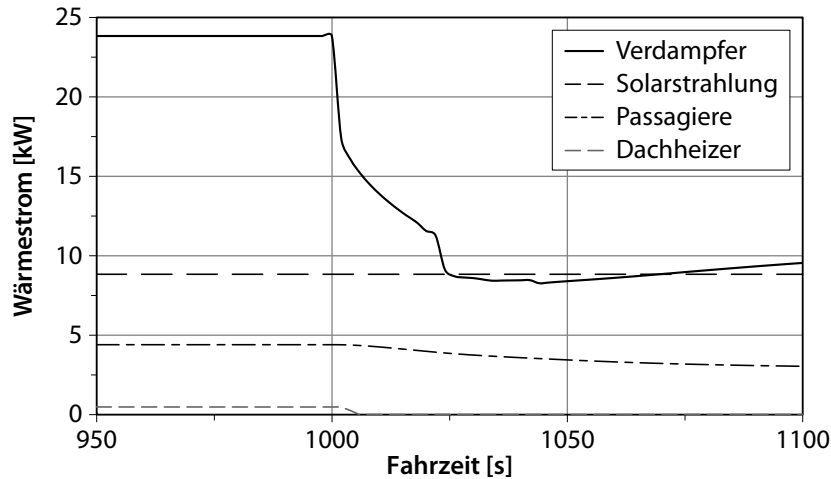


Abbildung 6.6.: Während der Kosimulation auftretende Wärmeströme: Kälteleistung des Verdampfers, einfallende Solarstrahlung sowie von Passagieren und Dachheizer abgegebene Wärme.

Abbildung 6.7 ist ebenfalls zu entnehmen, dass die zeitliche Verzögerung der Systemantwort – die Dynamik des Systems – von der gewählten Synchronisationsrate abhängt. Der stationäre Endwert (nicht dargestellt) ist jedoch bei allen Simulationen identisch.

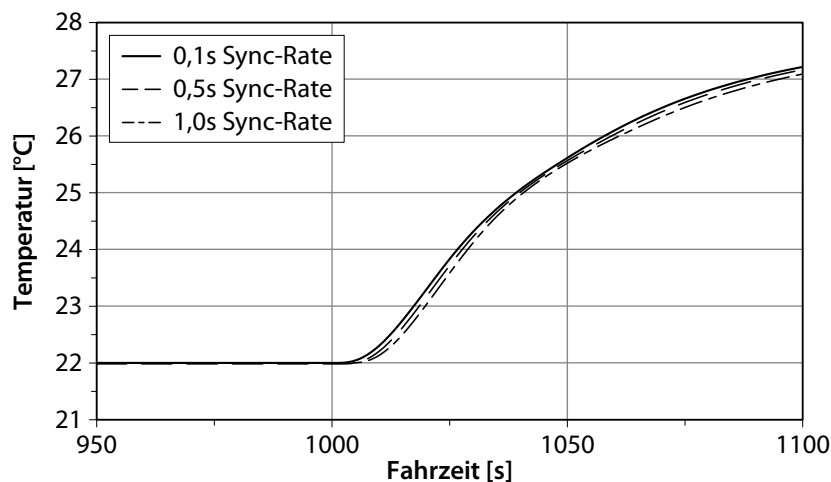


Abbildung 6.7.: Temperaturverlauf in der dritten Innenraumsektion (vgl. Kapitel 6.2.3) bei unterschiedlichen Synchronisationsraten.

Im Gegensatz zu der Innenraumtemperatur ist bei der im Gesamtkältekreislauf enthaltenen Kältemittelmenge eine Abweichung des stationären Endwerts festzustellen. Bei allen durchgeführten Simulationen erzeugt das Auskuppeln durch die starke Änderung der Verdichter-

drehzahl eine Änderung der Kältemittelmasse. Die Änderung ist umso größer, je seltener die Daten abgeglichen werden, je größer also die Synchronisationsrate gewählt ist: 0,12 % bei 0,1 s, 0,61 % bei 0,5 s und 1,25 % bei 1,0 s (vgl. Abbildung 6.8).

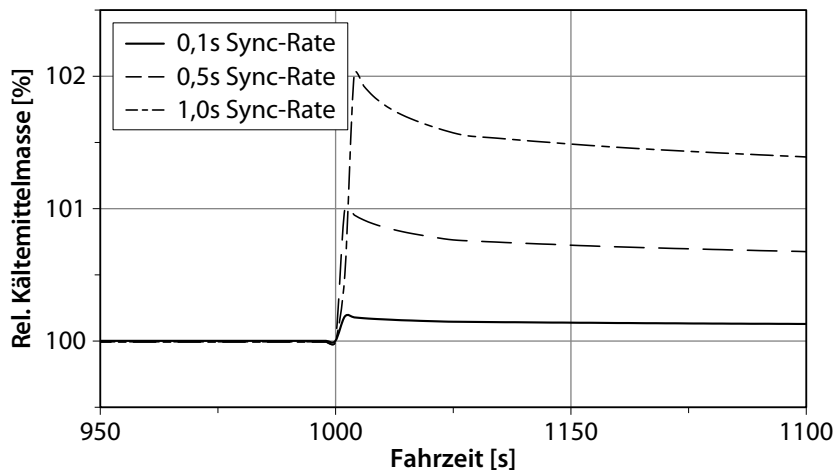


Abbildung 6.8.: Änderung der Kältemittelmasse bei unterschiedlichen Synchronisationsraten im gesamten Kältekreislauf. Als 100 % wird die Kältemittelmasse vor Beginn des Ausrollens bei 1000 s definiert.

Um den Einfluss dieses Effekts zu minimieren, wird festgelegt, dass die beiden Teilmodelle des Kältekreislaufs unabhängig von den übrigen angekoppelten Teilsystemmodellen mit einer Austauschrate von 0,1 s zu koppeln sind.

#### 6.3.2. Tunnelfahrt

Dieses Szenario bildet die Einfahrt in einen Tunnel nach. Innerhalb des Tunnels werden diffuse und direkte Solarstrahlung von den außen herrschenden  $750 \text{ W/m}^2$  bzw.  $165 \text{ W/m}^2$  auf  $0 \text{ W/m}^2$  gesenkt. Für die Außentemperatur werden drei Fälle betrachtet: gleichbleibend bei  $35^\circ\text{C}$ , ansteigend auf  $40^\circ\text{C}$  und abfallend auf  $30^\circ\text{C}$ . Die Beladung der Luft mit Wasser wird konstant gehalten; bei einer relativen Luftfeuchte von 25 % außerhalb des Tunnels ergeben sich somit für die letzten beiden Temperaturen relative Luftfeuchten von 19,1 % bzw. 33,1 %. Die Fahrgeschwindigkeit wird auf konstant  $50 \text{ km/h}$  definiert. Zur Untersuchung des Einflusses der Synchronisationsrate bei diesem Testszenario werden Simulationen mit Austauschintervallen von 0,1 s, 0,5 s, 1,0 s und 2,0 s durchgeführt. Die zwei Teilmodelle des Kältekreislaufs (vgl. Kapitel 6.2.1) werden mit einer maximalen Kopplungsrate von 1,0 s gekoppelt.

Abbildung 6.10 zeigt für die Simulationen mit einer Austauschrate von 0,1 s den vom Verdampfer übertragenen Wärmestrom, die in den Innenraum einfallende Solarstrahlung sowie die von Passagieren und Dachheizer abgegebenen Wärme. In Abbildung 6.11 sind für die unterschiedlichen Synchronisationsraten die Verläufe der Lufttemperatur in der dritten Innenraumsektion dargestellt. In diesen Temperaturverläufen ist für alle Umgebungsbedingungen ein identisches

Systemverhalten zu erkennen: Ausgehend von dem Verlauf bei einer Synchronisationsrate von  $0,1\text{ s}$  ist mit steigender Synchronisationsrate auch ein Anstieg der Neigung zu Schwingungen zu beobachten – bei einer Austauschrate von  $2,0\text{ s}$  schwingt die Innenraumtemperatur sogar dauerhaft. Dieses Verhalten kann durch Verwendung der Extrapolation (vgl. Kapitel 3.1.2) beeinflusst werden. Abbildung 6.9 zeigt zusätzlich zu den Verläufen bei  $0,1\text{ s}$  und  $2,0\text{ s}$  Austauschrate ohne Extrapolation auch den Verlauf, der sich bei einer Austauschrate von  $2,0\text{ s}$  unter Verwendung kubischer Extrapolation ergibt.

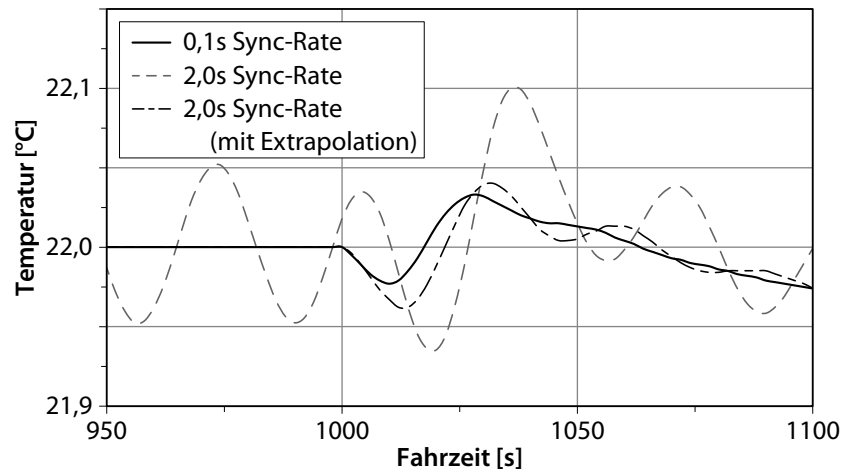
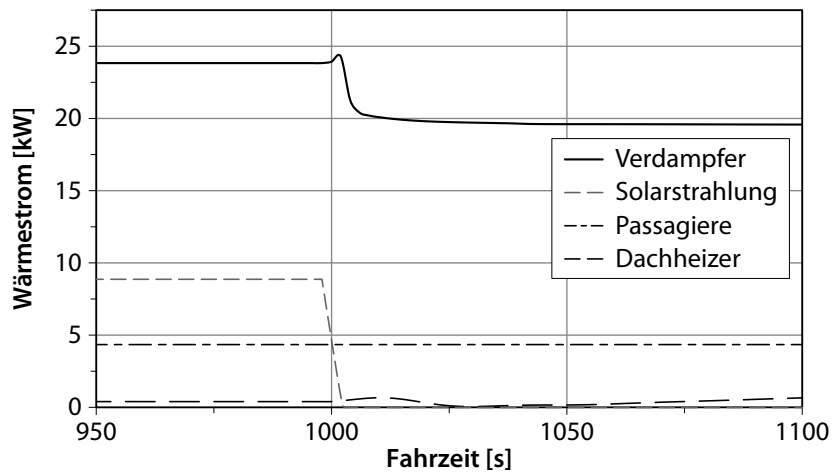
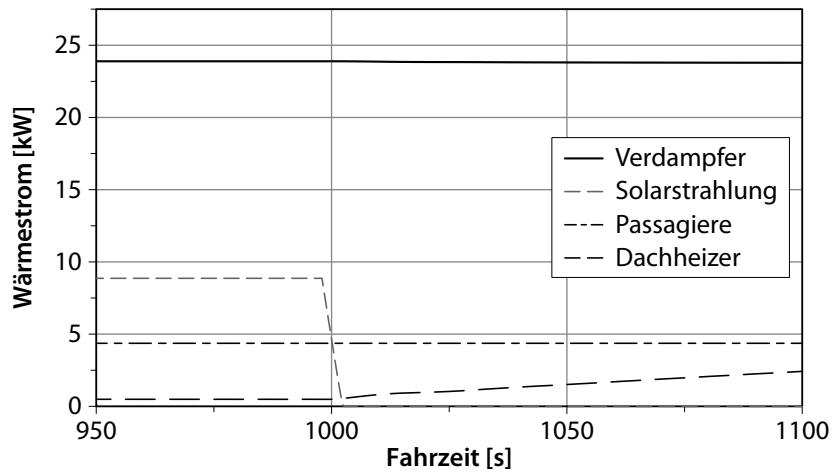


Abbildung 6.9.: Einfluss der Extrapolation auf den Temperaturverlauf in der dritten Innenraumsektion bei einer Synchronisationsrate von  $2,0\text{ s}$  und einem Sprung der Außentemperatur auf  $40^{\circ}\text{C}$  bei Tunneleinfahrt (mit Referenzkurve bei  $0,1\text{ s}$  Synchronisationsrate).

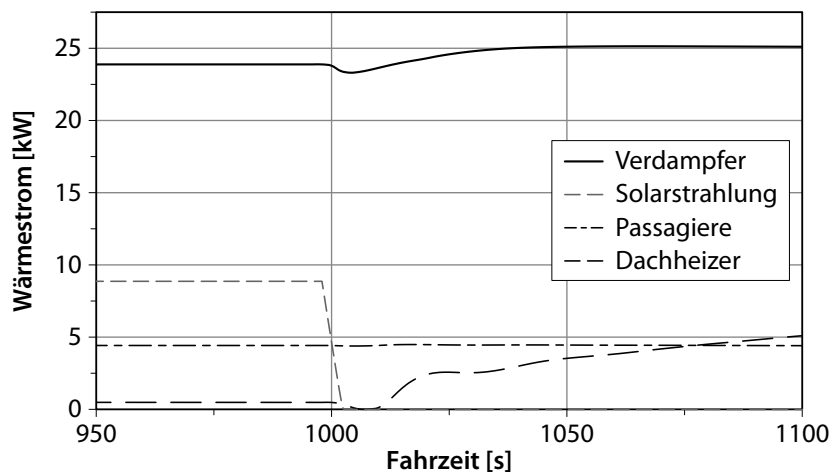
Neben der erzielbaren Ergebnisgenauigkeit ist die für die Simulation benötigte Rechenzeit ein weiteres Kriterium beim Festlegen der Synchronisationsraten der einzelnen Teilsysteme. Abbildung 6.12 zeigt die einzelnen Rechenzeiten der Teilsysteme für die vier Synchronisationsraten. Da die drei untersuchten Außentemperaturen keinen Einfluss zeigen, ist hier nur jeweils ein Diagramm dargestellt.



(a) Sprung der Außentemperatur auf 40°C.



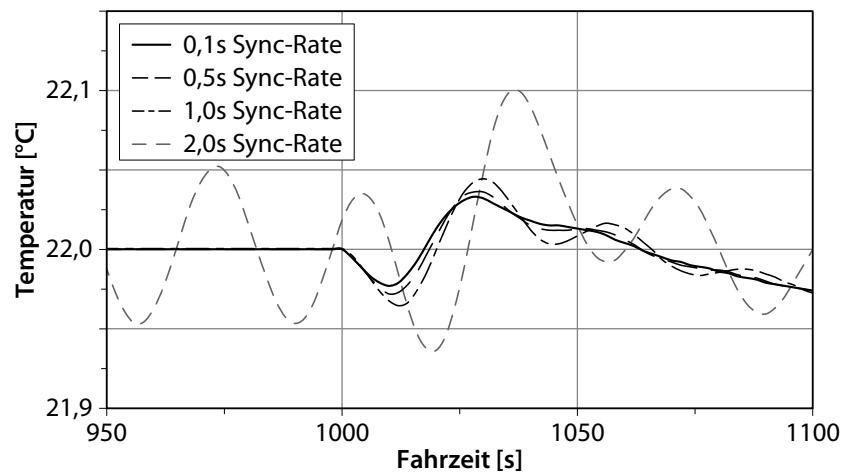
(b) Außentemperatur gleichbleibend bei 35°C.



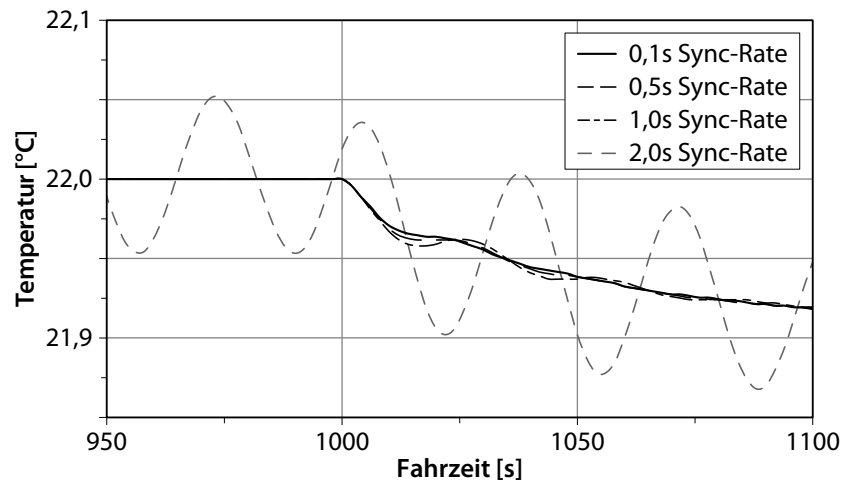
(c) Sprung der Außentemperatur auf 30°C.

Abbildung 6.10.: Kälteleistung des Verdampfers, vom Innenraum aufgenommene Solarstrahlung sowie von Passagieren und Dachheizer abgegebene Wärme bei einer Synchronisationsrate von 0,1 s.

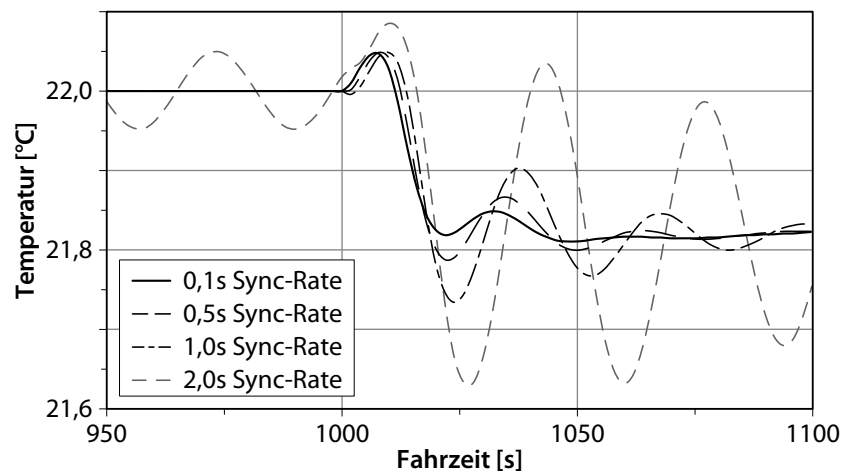




(a) Sprung der Außentemperatur auf 40°C.



(b) Außentemperatur gleichbleibend bei 35°C.

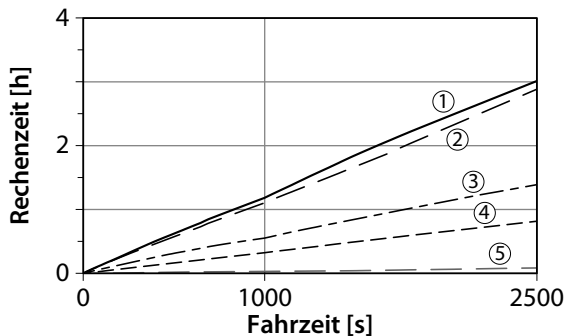


(c) Sprung der Außentemperatur auf 30°C.

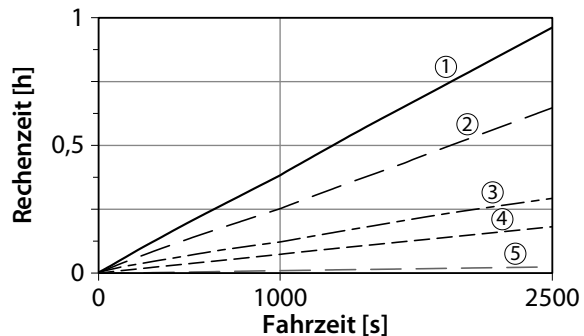
Abbildung 6.11.: Temperaturverlauf in der dritten Innenraumsektion bei unterschiedlichen Synchronisationsraten.

- ① Kühl-/Heizkreislauf      ③ Kältekreislauf (Verdampfer & Verdichter)      ⑤ Längsdynamik  
 ② Innenraum      ④ Kältekreislauf (Gaskühler & Ventil)

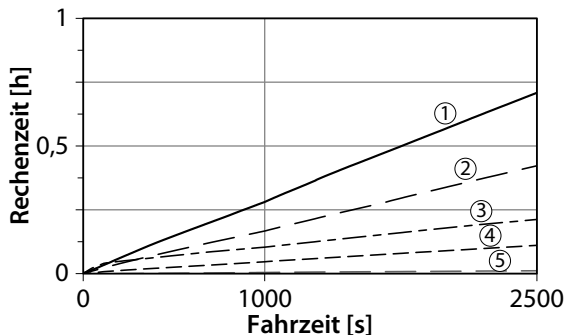
Synchronisation: 0,1s



Synchronisation: 0,5s



Synchronisation: 1,0s



Synchronisation: 2,0s

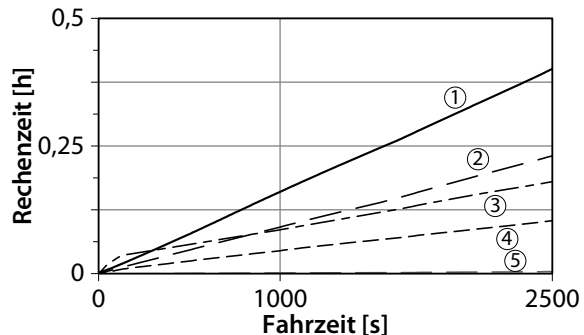


Abbildung 6.12.: Für die einzelnen Teilsysteme benötigte Rechenzeiten bei Synchronisationsraten von 0,1 s, 0,5 s, 1,0 s und 2,0 s.

In Kapitel 6.3.1 wurde festgelegt, dass die Teilsysteme des Kältekreislaufs (Nummern 3 und 4 in Abbildung 6.12) mit einer Frequenz von 0,1 s zu synchronisieren sind. Für die übrigen Systeme soll die Synchronisationsrate so gewählt werden, dass bei ausreichend hoher Genauigkeit die für die Kältekreislaufmodelle benötigte Rechenzeit nicht überschritten wird. Auf Grund der niedrigen für die Simulation der Längsdynamik benötigten Rechenzeit kann auch dieses System mit 0,1 s synchronisiert werden. Da die Berechnung der Modelle für den Kühl-/Heizkreislauf und den Innenraum die meiste Zeit benötigt (vgl. Abbildung 6.12, Nr. 1 und 2), ist es sinnvoll, für diese Systeme eine Synchronisationsrate von 0,5 s zu wählen. Die sich für eine solche Simulation mit unterschiedlichen Synchronisationsraten ergebenden Rechenzeiten zeigt Abbildung 6.13. Diese entsprechen jeweils den in Abbildung 6.12 dargestellten Verläufen, sodass hierdurch keine weiteren Kopplungseffekte eingeführt werden.

## 6.4. Reise Lissabon – Madrid

In diesem Kapitel werden die in den vorangegangenen Abschnitten vorgestellten Modelle verwendet, um eine energetische Bewertung des thermischen Systems eines Reisebusses zu treffen.

- ① Kühl-/Heizkreislauf      ③ Kältekreislauf (Verdampfer & Verdichter)      ⑤ Längsdynamik  
 ② Innenraum      ④ Kältekreislauf (Gaskühler & Ventil)

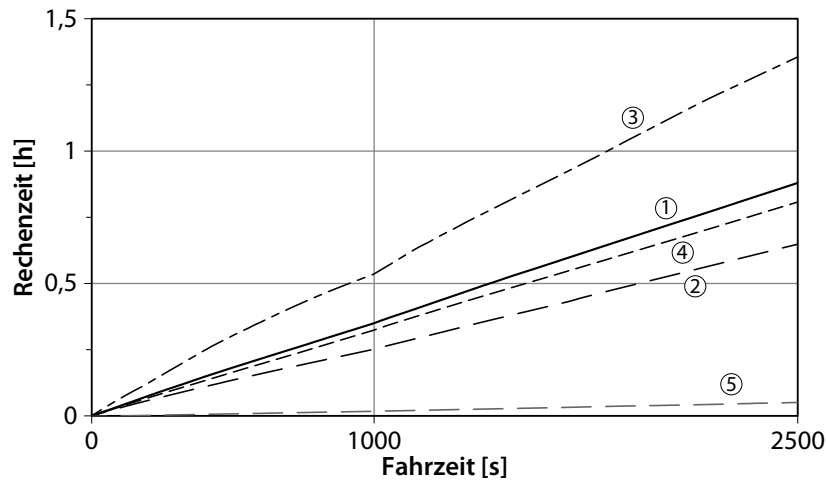


Abbildung 6.13.: Rechenzeiten bei einer Synchronisationsrate von 0,1 s für die Kältekreislaufmodelle sowie das Längsdynamikmodell und eine Synchronisationsrate von 0,5 s für Kühl-/Heizkreislauf- und Innenraummodell.

Üblicherweise werden für die energetische Betrachtung von Fahrzeugen genormte Fahrzyklen verwendet, um eine Vergleichbarkeit der erhaltenen Ergebnisse zu gewährleisten. Während für verschiedene Arten von Fahrzeugen diverse genormte Testzyklen existieren – Samuel u. a. [2002] und Neudorfer u. a. [2006] geben einen Überblick über die für PKW definierten Zyklen, Barlow u. a. [2009] stellen kurz Zyklen dar, die für PKW, Motorräder, LKW und Stadtbusse entwickelt wurden –, ist für Reisebusse kein genormter Fahrzyklus bekannt. Neben den Testzyklen werden heutzutage verstärkt virtuelle Fahrten simuliert, um die realen Einsatzbedingungen der Fahrzeuge besser nachzubilden [Daimler AG 2007].

Im Folgenden wird eine virtuelle Fahrt des Reisebusses von Lissabon (Portugal) nach Madrid (Spanien) simuliert. Abfahrts- und Zielort sowie Datum und Reisezeit sind dabei so ausgewählt, dass Fahr- und Umgebungszustände garantiert sind, bei denen eine Abkühlung des Innenraums durch die Klimaanlage notwendig ist. Besonderes Augenmerk wird hierbei auf die Energieeffizienz der Klimaanlage gelegt, die über die benötigte Antriebsleistung des Kältemittelverdichters bewertet wird. Betrachtet werden drei Strategien zur Regelung der Innenraumtemperatur:

### 1. Temperaturregelung durch Gegenheizung

In Anlehnung an die gängige Praxis bei mit dem Kältemittel R-744 betriebenen Kältekreisläufen wird der Öffnungsquerschnitt des Expansionsventils so geregelt, dass sich der energieoptimale Hochdruck einstellt (vgl. Kapitel 6.2.1). Da hierdurch ggf. mehr Kälteleistung erzeugt wird, als für die Abkühlung des Innenraums notwendig ist, wird durch den Dachheizer des Kühl-/Heizkreislaufs die Luft wieder aufgeheizt, bevor sie in den Innenraum geleitet wird.

**2. Temperaturregelung durch Zylinderbankabschaltung und Gegenheizung**

Zusätzlich zur 1. Regelstrategie wird das Hubvolumen des Kältemittelverdichters über die Abschaltung einer der beiden Zylinderbänke geregelt. Hierdurch kann die vom Kältekreislauf zur Verfügung gestellte Kälteleistung in zwei Stufen eingestellt werden. Die verwendete Regelung sieht vor, dass bei Außenlufttemperaturen unter  $35^{\circ}\text{C}$  der Verdichter mit halbem Hubvolumen betrieben wird.

**3. Temperaturregelung durch Expansionsventil und Zylinderbankabschaltung**

Bei dieser Regelung erfolgt die Bestimmung des EXV-Öfnungsquerschnitts nicht in Abhängigkeit vom Kältemittelhochdruck, sondern von der Innenraumtemperatur. Ziel ist die Vermeidung der Gegenheizung, indem durch den Kältekreislauf nicht mehr als die benötigte Kälteleistung bereitgestellt wird. Auch in diese Regelungsstrategie wird die Zylinderbankabschaltung einbezogen.

Zusätzlich wird die Umlufterate in den Stufen 0%, 25%, 50%, 75% und 100% geregelt. Wie bei real implementierten Systemen wird die Umlufterate in Abhängigkeit von der Differenz zwischen Innenraum- und Außentemperatur bestimmt. Abbildung 6.14 zeigt als Beispiel die sich ergebende Umlufterate bei Verwendung der Regelung durch Gegenheizung.

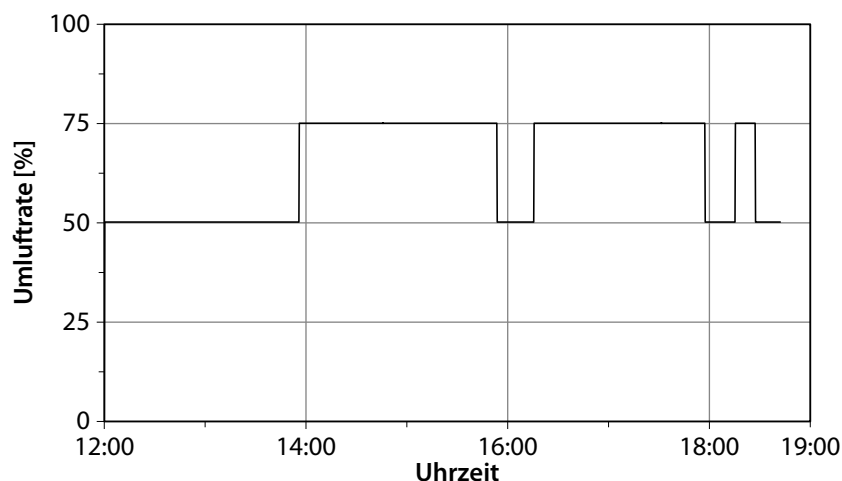


Abbildung 6.14.: Umlufterate während der virtuellen Fahrt, geregelt in Abhängigkeit von der Differenz zwischen Innenraum- und Außenlufttemperatur.

Zur Bestimmung des Fahrzustands wird die Fahrtroute von Lissabon nach Madrid mit Hilfe eines Routenplaners erstellt. Als Sollgeschwindigkeit des Reisebusses während der Simulation wird die im Routenplaner hinterlegte Höchstgeschwindigkeit für die befahrene Straße verwendet (vgl. Abbildung 6.15).

Zwischen 13:00 Uhr und 13:45 Uhr ist eine Stauumfahrung eingefügt, um das Verhalten des Gesamtsystems bei einer Variation der Fahrgeschwindigkeit betrachten zu können. Hierbei biegt der Reisebus von der Autobahn auf kleinere Straßen ab und fährt schließlich wieder auf die Autobahn auf. Dies ist auch in der Fahrtrichtung des Busses wiederzufinden, die in Abbildung 6.16 dargestellt ist. Abbildung 6.17 zeigt die Fahrtrichtung in Polarkoordinaten.

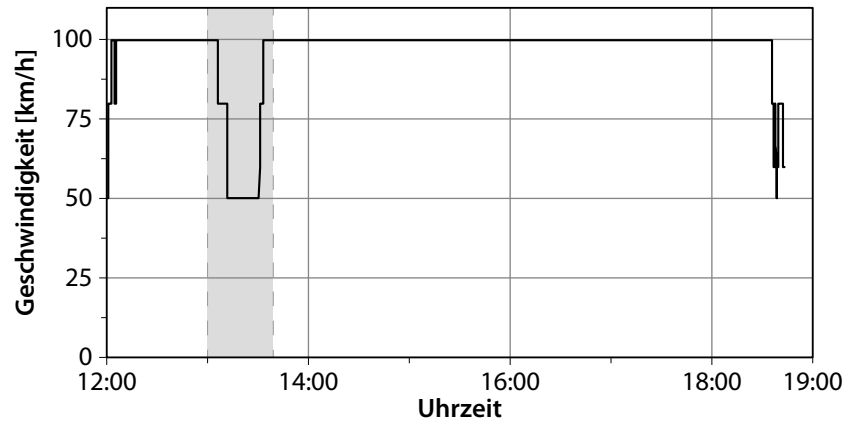


Abbildung 6.15.: Geschwindigkeitsvorgabe des Reisebusses (zulässige Höchstgeschwindigkeit auf den befahrenen Straßen). Hervorgehoben: Stauumfahrung.

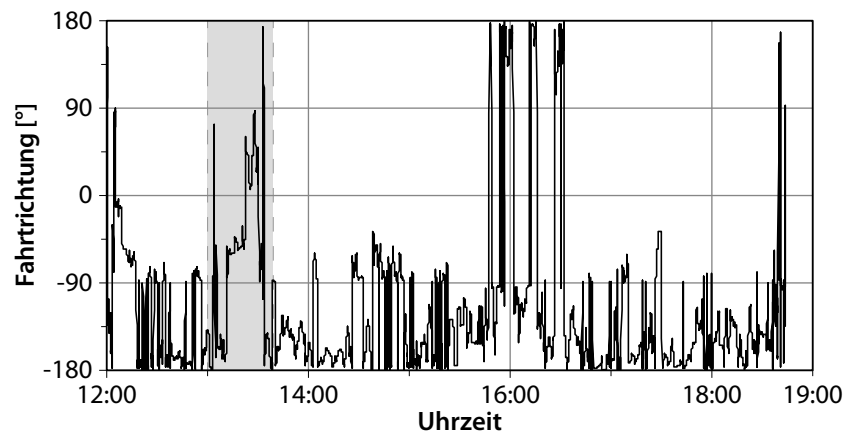


Abbildung 6.16.: Fahrtrichtung des Reisebusses ( $0^\circ$  = Süden,  $90^\circ$  = Westen,  $\pm 180^\circ$  = Norden,  $-90^\circ$  = Osten). Hervorgehoben: Stauumfahrung.

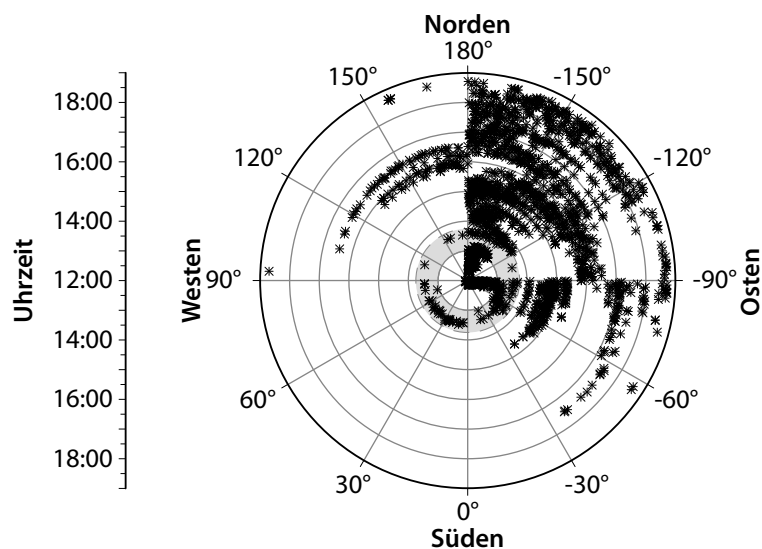


Abbildung 6.17.: Fahrtrichtung des Reisebusses in Polarkoordinaten. Hervorgehoben: Stauumfahrung.

Während der Fahrt wird der Umgebungszustand mit Hilfe eines eigens hierfür erstellten Programms berechnet. Dem Programm liegen an den in Abbildung 6.18 dargestellten Stützstellen stündlich aufgelöste Informationen über das lokale Klima in Form eines charakteristischen Jahres in Anlehnung an das Testreferenzjahr zu Grunde. Zu jedem Zeitpunkt der virtuellen Fahrt werden die Klimainformationen aus den Daten der nächstliegenden Stützstellen zur aktuellen und nächsten vollen Stunde interpoliert.



Abbildung 6.18.: Stützstellen für die Berechnung des Umgebungszustands (+) und geplante Reiseroute. Hintergrundgrafik: GPS Visualizer [2011].

Abbildung 6.19 zeigt die Tagesgänge der Temperaturen von Start- und Zielort sowie die Außenlufttemperatur während der Fahrt als Ergebnis der oben beschriebenen Interpolation.

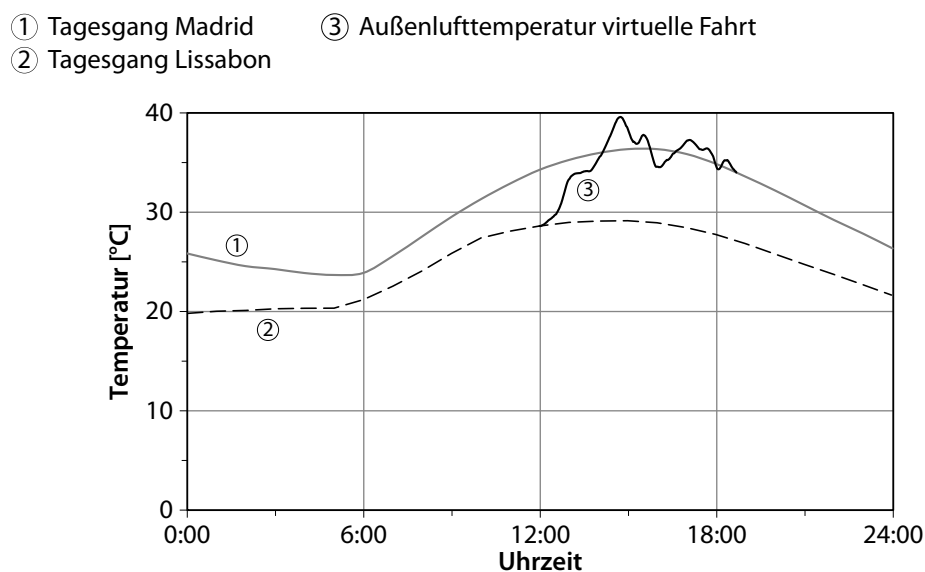


Abbildung 6.19.: Verläufe der Temperaturen am Startort Lissabon, am Zielort Madrid und der Außenlufttemperatur während der virtuellen Fahrt.

Die Abbildung zeigt zum einen, dass die übliche Annahme einer konstanten Außentemperatur nur für sehr kurze Testfahrten statthaft ist, zum anderen, dass Gelände, Ort und Uhrzeit bei der Festlegung der Randbedingungen berücksichtigt werden sollten.

Bei einer Regelung der Innenraumtemperatur durch Gegenheizen ergeben sich die in Abbildung 6.20 dargestellten Verläufe für die Temperaturen entlang des Luftpfads. Die Lufttemperatur am Eintritt des Verdampfers (Kurve 2) stellt sich in Folge der Mischung von Außenluft (Kurve 1) und als Umluft zurückgeführter Innenraumluft (Kurven 3) ein. Nach dem nachfolgenden Heizungswärmeübertrager (Kurve 4) wird die Luft in den Innenraum geleitet. Der Einfluss der Umluftregelung (vgl. Abbildung 6.14) lässt sich an Hand des Temperaturverlaufs der Luft am Verdampfereintritt (Kurve 2) erkennen. Die für den thermischen Komfort der Passagiere relevanten Temperaturen der vier Innenraumsektionen werden auf Grund der Regelung durch Gegenheizen mit dem Heizungswärmeübertrager (vgl. Kapitel 6.2.2) nahezu konstant gehalten.

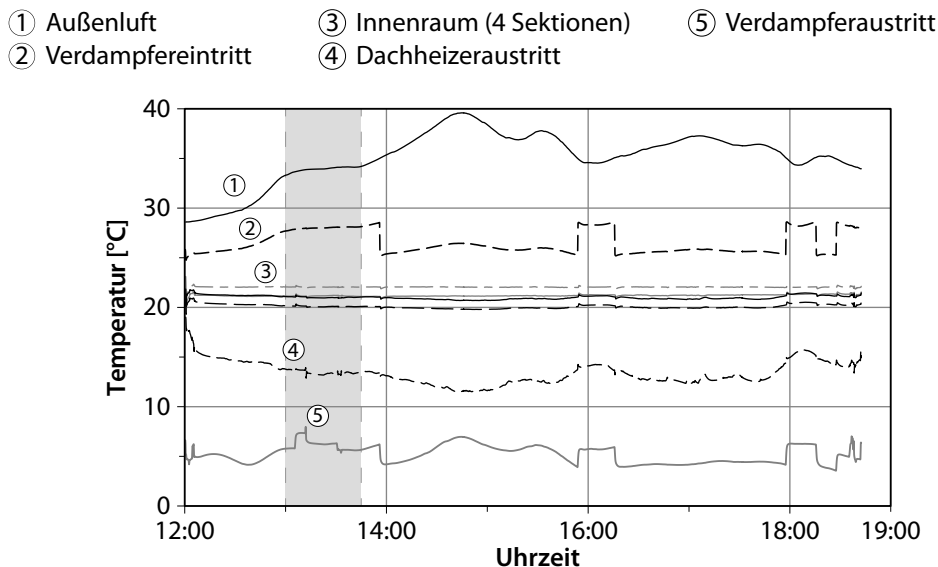


Abbildung 6.20.: Verläufe der Temperaturen im Luftpfad bei Regelung der Innenraumtemperatur durch Gegenheizen. Hervorgehoben: Stauumfahrung.

Abbildung 6.21 zeigt die Wärmeströme der Wärmeübertrager im Luftpfad (Verdampfer und Dachheizer), die solare Einstrahlung und die von den Passagieren abgegebene Wärme. Abhängig von der Ausrichtung des Reisebusses zur Sonne ändert sich die Fensterfläche, durch die direkte Solarstrahlung in den Innenraum eintritt. Aus diesem Grund zeigt der Betrag der solaren Einstrahlung eine ebenso hohe Dynamik wie die Fahrtrichtung (vgl. Abbildung 6.16). In Abbildung 6.21 sind die Auswirkungen der Stauumfahrung hervorgehoben dargestellt. Der im hervorgehobenen Bereich erkennbare Ausschlag in der Verdampferleistung ergibt sich durch den Vorgang des Gangwechsels beim Abbremsen des Busses. Bei einer höheren Umluftrate sinkt aufgrund der niedrigeren Lufttemperatur am Verdampfereintritt der Verdampfungsdruck

und damit die Dichte des Kältemittels am Verdichtereintritt. Durch die sich hierdurch ergebende Verringerung des geförderten Kältemittelmassenstroms folgt die dargestellte niedrigere Kälteleistung des Verdampfers.

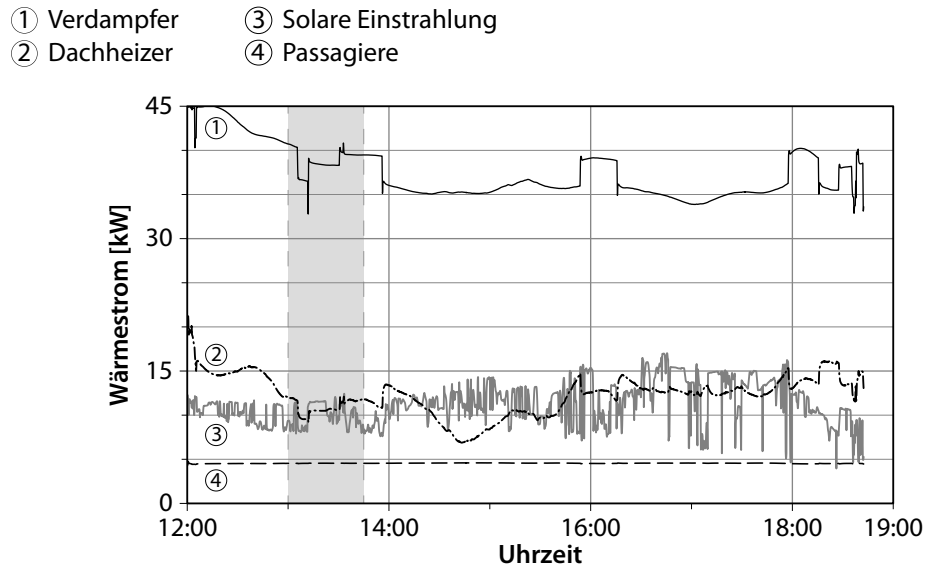


Abbildung 6.21.: Auswahl eingebrachter und abgeführter Wärmeströme bei Regelung der Innenraumtemperatur durch Gegenheizen. Hervorgehoben: Stauumfahrung.

Aus Abbildung 6.21 ist zu erkennen, dass für die Aufheizung der aus dem Verdampfer austretenden Luft bis zu  $15\text{ kW}$  aufgewendet werden müssen. Durch Abschalten einer Zylinderbank des Kältemittelverdichters bei einer Außenlufttemperatur von weniger als  $35^\circ\text{C}$  kann diese Heizleistung verringert werden, wie im Folgenden gezeigt wird. Durch die Zylinderbankabschaltung wird das genutzte Hubvolumen halbiert (siehe Abbildung 6.22), gleichzeitig sinkt die Antriebsleistung des Verdichters.

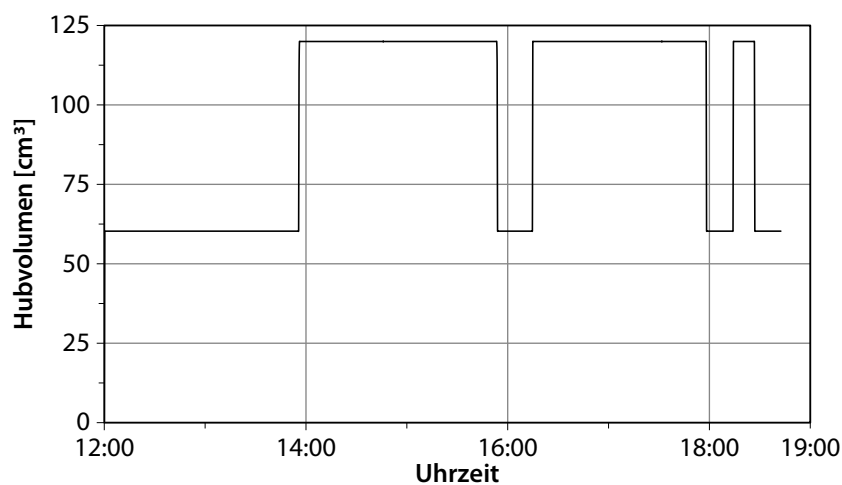


Abbildung 6.22.: Änderung des genutzten Kältemittelverdichterhubvolumens durch Zylinderbankabschaltung bei einer Außentemperatur unter  $35^\circ\text{C}$ .



Die sich bei Verwendung der Zylinderbankabschaltung einstellenden Temperaturen und Wärmeströme sind in Abbildung 6.23 bzw. Abbildung 6.24 dargestellt.

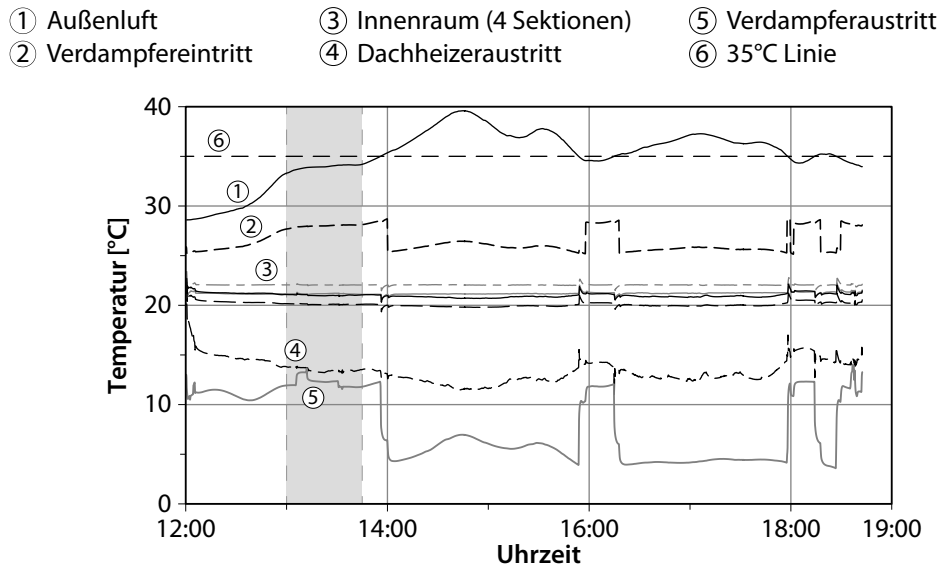


Abbildung 6.23.: Verläufe der Temperaturen im Luftpfad bei Abschaltung einer Zylinderbank des Kältemittelverdichters unterhalb von 35°C Außentemperatur. Hervorgehoben: Stauumfahrung.

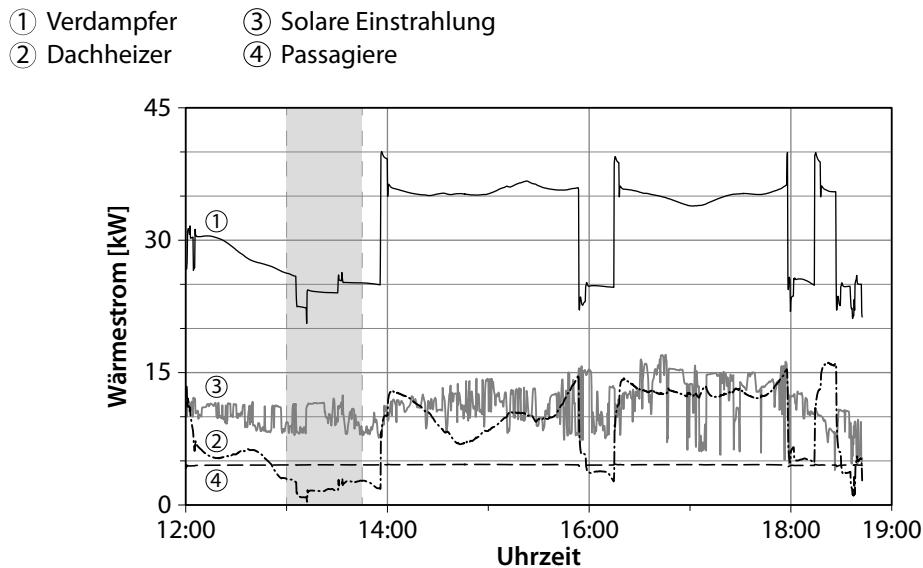


Abbildung 6.24.: Auswahl eingebrachter und abgeführter Wärmeströme bei Abschaltung einer Zylinderbank des Kältemittelverdichters unterhalb von 35°C Außentemperatur. Hervorgehoben: Stauumfahrung.

Eine Abweichung zu den in Abbildung 6.20 gezeigten Temperaturverläufen bei vollem Verdichterhubvolumen ergibt sich vor allem in der Lufttemperatur am Verdampferaustritt. Dies erklärt sich dadurch, dass die Temperatur am Verdampfereintritt identisch, die erzielbare Kälteleistung jedoch deutlich verringert ist (vgl. Abbildung 6.21, Nr. 1 und Abbildung 6.24, Nr. 1).

Die erkennbaren Schwankungen der Innenraumtemperatur zum Zeitpunkt der Zylinderbankabschaltung sind auf die schnelle Änderung der Kälteleistung zurückzuführen. Durch die unabhängige, aber etwa zeitgleich auftretende Umschaltung der Umlufterate werden diese Schwankungen weiter verstärkt.

Aus Abbildung 6.24 wird deutlich, dass zu jedem Zeitpunkt ausreichend Kälteleistung zur Verfügung steht, da auch bei halbiertem Verdichterhubvolumen jederzeit über den Heizungswärmeübertrager gegengeheizt wird. Die hierfür aufgewendete Heizleistung sinkt durch die Zylinderbankabschaltung von maximal 15 kW (vgl. Abbildung 6.21) auf maximal etwa 5 kW.

Der Nachteil der Regelung des Kältekreislaufs auf einen energieoptimalen Hochdruck mit anschließender Gegenheizung ist, dass mehr Kälteleistung erbracht wird, als zur Kühlung des Innenraums notwendig ist. Diese zusätzliche Leistung ist in dem zur Regelung des Kältekreislaufs verwendeten COP (Coefficient of Performance) nicht berücksichtigt, da sich dieser als Quotient aus Verdampfer- und Verdichterleistung berechnet und somit abgeschlossen auf den Kältekreislauf bezieht. Wird der COP anstatt auf die Verdampfer- auf die Nutzkälteleistung (also unter Berücksichtigung der Gegenheizung) bezogen, ergibt sich ein abweichender energieoptimaler Hochdruck bei einer höheren Gesamtenergieeffizienz, wie Abbildung 6.25 für einen exemplarischen Betriebspunkt illustriert.

- ① basierend auf Verdampferkälteleistung      a) Optimum COP Verdampferkälteleistung  
② basierend auf Nutzkälteleistung              b) Optimum COP Nutzkälteleistung

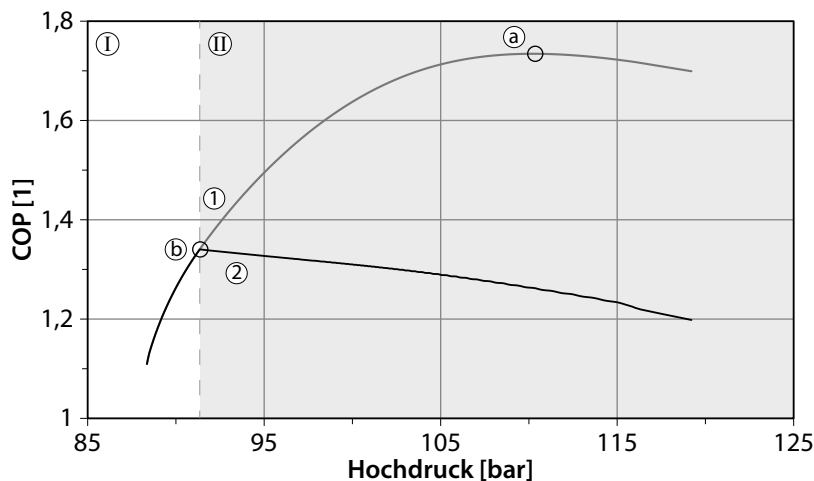


Abbildung 6.25.: Vergleich von auf Verdampfer- und Nutzkälteleistung basierendem COP für einen exemplarischen Betriebspunkt, aufgetragen über dem Kältemittelhochdruck. Bereich I: Inaktive Gegenheizung, Bereich II: Aktive Gegenheizung.

Es ist zu erkennen, dass die Berücksichtigung der Gegenheizung einen signifikanten Einfluss auf den COP hat. Das Optimum verschiebt sich in den Punkt, in dem die Verdampferleistung der benötigten Nutzkälteleistung entspricht, wodurch keine Gegenheizung benötigt wird. Für die Einstellung des Kältekreislaufs auf diesen Betriebspunkt muss die Regelung des EXV-Öffnungsquerschnitts angepasst werden. Abbildung 6.26 zeigt das Schema des angepassten

Kältekreislaufmodells (vgl. Abbildung 6.2), in dem an Stelle des Kältemittelhochdrucks die Innenraumtemperatur als Regelgröße für das Ventil verwendet wird.

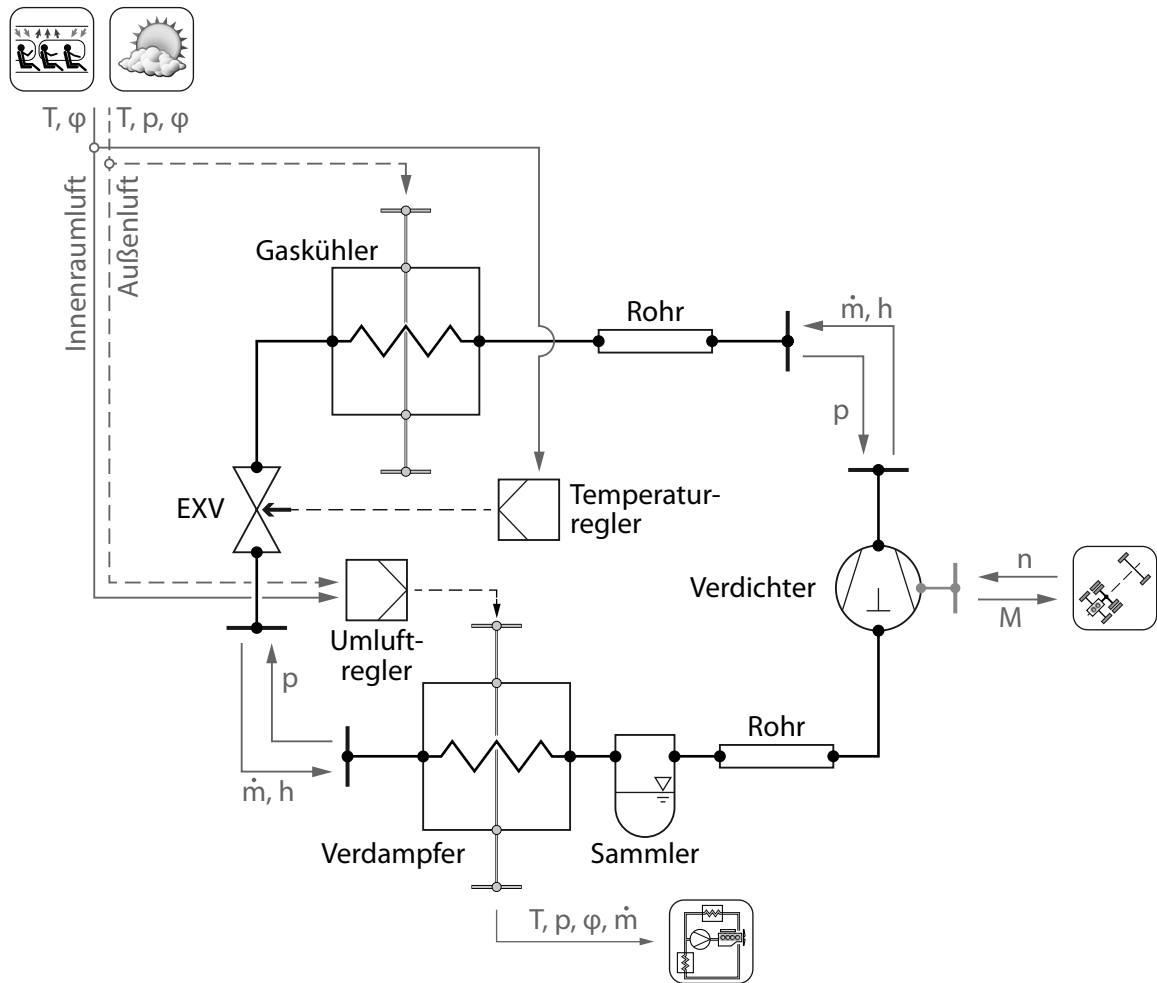


Abbildung 6.26.: Schema des Kältekreislaufmodells nach Abbildung 6.2 mit Regelung des EXV-Öffnungsquerschnitts über die Innenraumtemperatur an Stelle des Hochdrucks.

Abbildung 6.27 zeigt im ph-Diagramm den dynamischen Zwischenzustand des Kältekreislaufs unter den drei betrachteten Regelungsstrategien für den Zeitpunkt 15:00 Uhr der virtuellen Fahrt. Da die Umgebungstemperatur bei  $38,4^\circ\text{C}$  und somit oberhalb von  $35^\circ\text{C}$  liegt, arbeitet der Kältemittelverdichter bei allen Regelstrategien mit vollem Hubvolumen. Aus diesem Grund ergibt sich die Abweichung von Regelstrategie 3 zu den beiden anderen ausschließlich durch die geänderte Regelung des EXV. Während der Hochdruck der Kreisläufe 1 und 2 dem energieoptimalen Hochdruck entspricht, ist erkennbar, dass dies auf Kreislauf 3 nicht zutrifft (vgl. Inokuty [1928]). Zur Anhebung des Hochdrucks von Kreislauf 3 bei gleichbleibender Kälteleistung am Verdampfer wäre neben einer Nachregelung des EXV eine Absenkung des Kältemittelmassenstroms notwendig. Dies ist etwa über eine Änderung der Verdichterdrehzahl oder des Verdichterhubvolumens möglich; da der im Rahmen dieser Arbeit betrachtete Verdichtertyp solche Änderungen nicht oder nicht ausreichend erlaubt, stehen diese Optionen hier nicht zur Verfügung und sollten in einer weiterführenden Arbeit betrachtet werden.

Regelung der Innenraumtemperatur durch:

- ① Gegenheizung
- ② Zylinderbankabschaltung und Gegenheizung
- ③ Expansionsventil, mit Zylinderbankabschaltung

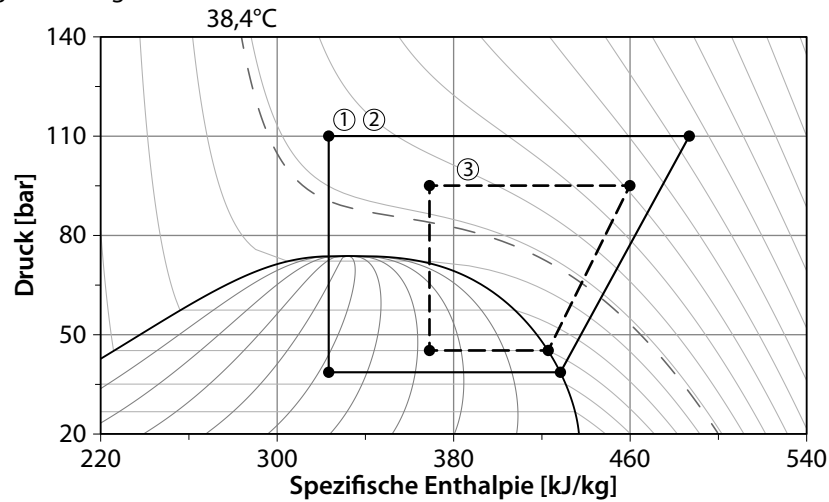


Abbildung 6.27.: Dynamischer Zwischenzustand des Kältekreislaufs um 15:00 Uhr bei unterschiedlichen Strategien zur Regelung der Innenraumtemperatur.

Zur Bewertung der möglichen Energieeinsparung zeigt Abbildung 6.28 die Antriebsleistung des Kältemittelverdichters für die drei vorgestellten Simulationen mit Regelung der Innenraumtemperatur mit Gegenheizung (1), mit zusätzlicher Zylinderbankabschaltung (2) und Regelung mit Expansionsventil und Zylinderbankabschaltung (3).

Regelung der Innenraumtemperatur durch:

- ① Gegenheizung
- ② Zylinderbankabschaltung und Gegenheizung
- ③ Expansionsventil, mit Zylinderbankabschaltung

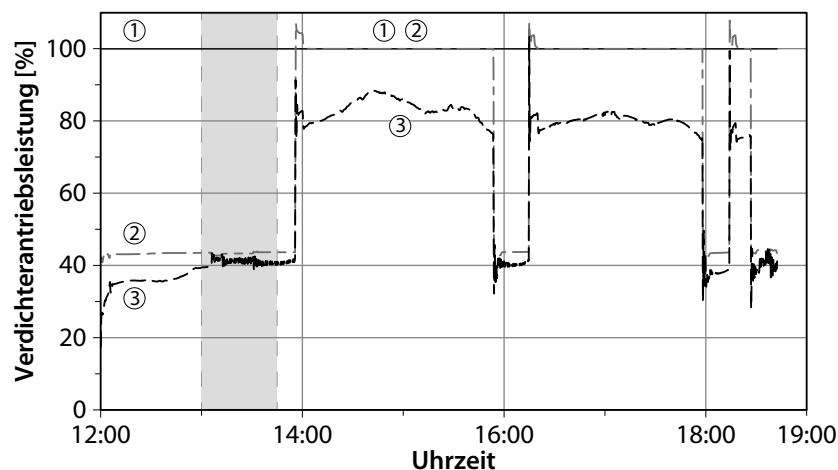


Abbildung 6.28.: Antriebsleistung des Kältemittelverdichters bei unterschiedlichen Strategien zur Regelung der Innenraumtemperatur. 100 % entspricht der Antriebsleistung bei Regelung über Gegenheizung. Hervorgehoben: Stauumfahrung.

Wie Abbildung 6.28 zu entnehmen ist, kann durch die Regelung mit Expansionsventil zu jedem Zeitpunkt Antriebsenergie eingespart werden. Bei den verwendeten Modellen ergibt sich für die betrachtete virtuelle Testfahrt eine Reduzierung der insgesamt benötigten Verdichterantriebsenergie – dem Integral der Antriebsleistung – von etwa 23 % durch die Zylinderbankabschaltung und von etwa 36 % durch die zusätzliche Regelung mit Expansionsventil, jeweils bezogen auf die Regelung durch Gegenheizung (vgl. Abbildung 6.29).

Regelung der Innenraumtemperatur durch:

- ① Gegenheizung
- ② Zylinderbankabschaltung und Gegenheizung
- ③ Expansionsventil, mit Zylinderbankabschaltung

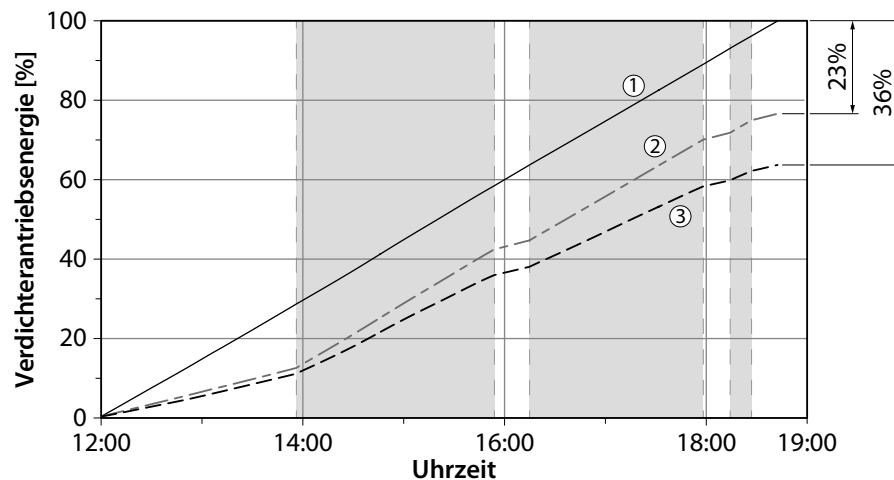


Abbildung 6.29.: Antriebsenergie des Kältemittelverdichters bei unterschiedlichen Strategien zur Regelung der Innenraumtemperatur. 100 % entspricht der insgesamt benötigten Antriebsenergie bei Regelung über Gegenheizung. Hervorgehoben: Betrieb mit vollem Verdichtervolumen.



## 7. Zusammenfassung

Bei der energetischen Betrachtung eines komplexen thermischen Systems ist die Einbeziehung der Wechselwirkungen zwischen den einzelnen Teilsystemen essentiell. Monolithische Modelle können häufig nicht alle Teilsysteme adäquat abbilden – viele der eingesetzten Simulationsprogramme sind auf eine technische Disziplin spezialisiert und enthalten oft nur rudimentäre Modelle für die angrenzenden Systeme. Eine Lösung dieses Problems stellt der Einsatz hybrider Simulation dar, bei der verschiedene Simulationstechniken in einem Verbund verwendet und unterschiedliche Modelle zu einem virtuellen Prototypen verknüpft werden können. Eine spezielle Art der hybriden Simulation ist die Kosimulation, bei der unterschiedliche, auf einen Teilaspekt spezialisierte Simulationsprogramme zu einer Gesamtsimulation verbunden werden. Hierbei ergibt sich ein zusätzlicher ökonomischer Vorteil dadurch, dass das für die einzelnen Simulationsprogramme bestehende Expertenwissen weiterhin genutzt werden kann. Durch die Weiternutzung bestehender Komponentenmodelle entfällt außerdem die zeitaufwändige Entwicklung und Validierung neuer Modelle.

Im Rahmen dieser Arbeit wird eine Kosimulationsumgebung entwickelt, die sich auf die Simulation thermischer Systeme konzentriert, jedoch auch für andere Fachgebiete verwendbar ist. Hierfür werden neue Methoden vorgestellt, um das Verhalten der gekoppelten Simulation in Richtung höherer Genauigkeit, Geschwindigkeit und Stabilität zu beeinflussen – neben der Synchronisation der Teilsystemsimulationen sind es vor allem numerische Effekte, die durch den Einsatz von Extrapolations- und Glättungsfunktionen sowie Randbedingungszeitreihen behandelt werden. Durch Schaffung der Möglichkeit, für die verschiedenen Teilsystemsimulationen unterschiedliche Synchronisationsraten zu verwenden, kann die jeweilige Austauschrate an die entsprechende Systemzeitkonstante angepasst und hierdurch das Systemverhalten und die Simulationsgeschwindigkeit optimiert werden. Mittels Extrapolation wird die Abweichung des Werteverlaufs im empfangenden Modell zum Werteverlauf im sendenden Modell verringert. Durch Glättung wird Stetigkeit im Werteverlauf und in den ersten beiden zeitlichen Ableitungen geschaffen, wodurch dem numerischen Gleichungslöser des verwendeten Simulationsprogramms die Berechnung des Systems erleichtert wird. Mit Einführung von Randbedingungszeitreihen wird zudem eine Möglichkeit präsentiert, die volle Dynamik fest vorgegebener Randbedingungen wie einem Fahrprofil zu erhalten (auch wenn diese über der verwendeten Synchronisationsrate liegt) und an die verschiedenen Teilsystemsimulationen zu übergeben. Für unterschiedliche technische Disziplinen wird eine Übersicht gegeben, welche Variablen für die Kopplung der Teilsysteme auszuwählen sind und wie z.B. durch das Einführen zusätzlicher Wärmekapazitäten und Wärmeleitwiderstände das Verhalten der Kosimulation eines

---

thermischen Systems beeinflusst werden kann. Die Auswirkungen der erwähnten Maßnahmen werden anhand der verteilten Simulation eines Kältekreislaufs untersucht. Hierfür werden einzelne Komponentenmodelle verwendet, die jeweils validiert sind und mittels Kosimulation zu einem Gesamtmodell zusammengefügt und berechnet werden. Hierbei zeigt sich die Wichtigkeit der Abstimmung der Kosimulationseinstellungen (Synchronisationsrate, Extrapolation, Glättung, evtl. zusätzliche Widerstände und Kapazitäten) auf das zu simulierende System.

Die gesamte Kosimulationsumgebung ist plattformübergreifend umgesetzt. Da die Kommunikation zwischen den beteiligten Simulationsprogrammen über Netzwerk erfolgt, können mehrere Computer und auch unterschiedliche Betriebssysteme für die Simulation genutzt werden. Bei der Entwicklung der Kosimulationsumgebung sind die einfache Anbindung weiterer Simulationsprogramme sowie die zentrale Konfiguration und Durchführung der Kosimulation im Vordergrund stehende Ziele.

Zur Strukturierung des Aufbaus und der Durchführung einer Kosimulation wird in dieser Arbeit ein Arbeitsprozess vorgestellt. Dieser berücksichtigt die Organisationsstruktur, wie sie in großen Firmen heutzutage üblich ist, und zielt auf die fachgruppenübergreifende Zusammenarbeit. Hierbei werden die Rollen des Simulationsverantwortlichen, des Modellentwicklers und des Simulationsanwenders definiert und diesen Rollen einzelne Aufgaben und Arbeitsschritte zugeordnet. Der Arbeitsprozess ist als Minimalprozess zu verstehen und stellt somit den Ausgangspunkt für detailliertere Prozesse dar, die abgestimmt auf die Erfordernisse einzelner Firmen erstellt werden.

Als Abschluss wird in dieser Arbeit die Kosimulation des thermischen Gesamtsystems eines Reisebusses vorgestellt, wobei sich die Vorgehensweise an dem entwickelten Arbeitsprozess orientiert. Nach der Partitionierung des Gesamtsystems in die vier Teilsysteme Kältekreislauf (mit dem Kältemittel R-744), Kühl-/Heizkreislauf, Innenraum und Längsdynamik werden zwei zusätzliche Modelle definiert, die den Fahrzustand und den Umgebungszustand während der Simulation bestimmen. Durch Berücksichtigung der Wechselwirkungen zwischen diesen Teilsystemen wird eine energetische Bewertung ermöglicht. Zur Festlegung der Kosimulationseinstellungen des gekoppelten Gesamtmodells (Synchronisationsrate, Extrapolation, Glättung) werden Sprungfunktionen als Randbedingungen verwendet. Mittels Analyse der Systemantworten werden die Werte dieser einzelnen Parameter festgelegt. Mit den so konfigurierten Modellen wird eine virtuelle Fahrt des Reisebusses von Lissabon (Portugal) nach Madrid (Spanien) präsentiert. Reiseroute sowie Datum und Uhrzeit des Reisebeginns werden so gewählt, dass zu jedem Zeitpunkt eine Kühlung des Innenraums notwendig ist. Neben der an die gängige Praxis angelehnten Regelung des R-744-Kältekreislaufs auf einen energieoptimalen Hochdruck mit anschließender Gegenheizung werden in dieser Arbeit der Einfluss einer Zylinderbankabschaltung des Kältemittelverdichters unterhalb einer Außentemperatur von 35 °C sowie eine Regelung der Innenraumtemperatur über Zylinderbankabschaltung und das elektronische Expansionsventil des Kältekreislaufs, bei der eine Gegenheizung vermieden wird, in Hinblick auf die Gesamtenergieeffizienz untersucht. Die Regelstrategien werden mittels der im Gesamtsystem auftretenden Lufttemperaturen und der einzelnen Wärmeströme bewertet. Für



den betrachteten Anwendungsfall ergibt sich eine Reduzierung der Verdichterantriebsleistung von etwa 22 % durch die Zylinderbankabschaltung und von etwa 36 % durch die zusätzliche Regelung mit Expansionsventil, jeweils bezogen auf die Regelung durch Gegenheizung.



# A. Literaturverzeichnis

## Ackerl u. a. 2010

ACKERL, M. ; HAGER, J. ; PRENNINGER, K. ; WINTER, M.: Energy Management and Thermal Simulation of Hybrid commercial vehicles. In: NIEDERSACHSEN E.V., Intelligente T. d. (Hrsg.): *7th Symposium on Hybrid Vehicles, Electric Vehicles and Energy Management*, 2010. – ISBN 978-3-937655-24-6, S. 98 – 114

## Amdahl 1967

AMDAHL, G.: Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities. In: *AFIPS '67 (Spring): Proceedings of the April 18-20, 1967, spring joint computer conference* Bd. 30. New York, NY, USA : ACM, 1967. <http://dx.doi.org/10.1145/1465482.1465560>. S. 483 – 485

## Arnold 1998

ARNOLD, M.: *Fortschritts-Berichts VDI, Reihe 20*. Bd. 264: *Zur Theorie und zur numerischen Lösung von Anfangswertproblemen für differentiell-algebraische Systeme von höherem Index*. Düsseldorf : VDI Verlag, 1998

## Arnold 2004

ARNOLD, M.: Simulation Algorithms in Vehicle System Dynamics / Martin-Luther-Universität Halle-Wittenberg. 2004 (Reports of the Institute of Numerical Mathematics 2004,27). – Forschungsbericht

## Arnold 2006

ARNOLD, M.: Multi-Rate Time Integration for Large Scale Multibody System Models. In: EBERHARD, P. (Hrsg.) ; International Union of Theoretical and Applied Mechanics (Veranst.): *IUTAM Symposium on Multiscale Problems in Multibody System Contacts* International Union of Theoretical and Applied Mechanics, Springer Netherlands, 2006. – ISBN 978-1-4020-5980-3. [http://dx.doi.org/10.1007/978-1-4020-5981-0\\_1](http://dx.doi.org/10.1007/978-1-4020-5981-0_1). S. 1 – 10

## Arnold u. Schierz 2009

ARNOLD, M. ; SCHIERZ, T.: Effizienz und Robustheit numerischer Kopplungsalgorithmen im MODELISAR Co-Simulation-Interface. In: ELST, G. (Hrsg.) ; Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwurfsautomatisierung EAS (Veranst.): *Tagungsband ASIM-Treffen STS/GMMS 2009 und DASS 2009, Dresden, 05.03.-06.03.09* Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwurfsautomatisierung EAS, Fraunhofer IRB Verlag, März 2009. – ISBN 978-3-8167-7981-0

## Barlow u. a. 2009

BARLOW, T. ; LATHAM, S. ; MCCRAE, I. ; BOULTER, P.: *A reference book of driving cycles for use in the measurement of road vehicle emissions*. TRL Limited, 2009. – ISBN 978-1-84608-816-2

---

**Baron 2008**

*Kapitel 2.3.* In: BARON, K.: *Virtuelle Systementwicklung – Von der Anforderung zum Steuergerät*. 1. Vieweg + Teubner, 2008. – ISBN 978–3–8348–0345–0, S. 133 – 153

**Betz u. a. 2005**

BETZ, J. ; ANZENBERGER, T. ; KOB, T.: Entwicklungstendenzen der Wärmemanagementsimulation im Bereich Kühlung und Klimatisierung bei AUDI. In: STEINBERG, P. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs IV* Haus der Technik, Expert-Verlag, Dezember 2005 (Haus der Technik Fachbuch 43). – ISBN 978–3–8169–2478–4, S. 126 – 140

**Binder u. a. 2008**

*Kapitel 1.1.* In: BINDER, T. ; BOCK, T. ; DICK, M. ; KRIEGER, T. ; MÜLLER, T.: *Einsatz virtueller Techniken in der Produktentwicklung*. 1. Vieweg + Teubner, 2008. – ISBN 978–3–8348–0345–0, S. 1 – 29

**Boehr 2005**

BOEHR, C.: *Distributed Simulation in Industry*. Rotterdam, Netherlands, Erasmus University Rotterdam, Diss., Oktober 2005. – ISBN 978-3-86624-214-2, Available via <http://hdl.handle.net/1765/6925>

**Bryant 1977**

BRYANT, R.: *Simulation of Packet Communication Architecture Computer Systems*. Cambridge, MA, USA : Massachusetts Institute of Technology, 1977. – Forschungsbericht

**Busch 2006**

BUSCH, M.: *Entwicklung einer Simpack-Modelica/Dymola-Schnittstelle*, Martin-Luther Universität Halle-Wittenberg, Diplomarbeit, Oktober 2006

**Buss u. Jackson 1998**

BUSS, A. ; JACKSON, L.: Distributed simulation modeling: a comparison of HLA, CORBA, and RMI. In: *WSC '98: Proceedings of the 30th conference on Winter simulation*. Los Alamitos, CA, USA : IEEE Computer Society Press, 1998. – ISBN 0–7803–5134–7, S. 819 – 826

**Cellier u. a. 1996**

*Kapitel 1.2.3.1.* In: CELLIER, F. ; ELMQVIST, H. ; OTTER, M.: *Modeling from Physical Principles*. CRC Press, 1996, S. 99 – 108

**Chandy u. Misra 1979**

CHANDY, K. ; MISRA, J.: Distributed Simulation: A Case Study in Design and Verification of Distributed Programs. In: *Software Engineering, IEEE Transactions on SE-5* (1979), September, Nr. 5, S. 440 – 452. – ISSN 0098–5589

**Correia 2009**

CORREIA, C.: *Investigation of network coupling methods for thermal system simulations*. Braunschweig, Deutschland, Technische Universität Braunschweig, Diss., Oktober 2009

**Daimler AG 2007**

*Digitaler Prototyp – Vorsprung durch Berechnung*. Pressemitteilung Daimler AG, Januar 2007

**DeBenedictis u. a. 1991**

DEBENEDICTIS, E. ; GHOSH, S. ; YU, M.: A Novel Algorithm for Discrete-Event Simulation: Asynchronous Distributed Discrete-Event Simulation Algorithm for Cyclic Circuits Using a Dataflow Network. In: *Computer* 24 (1991), S. 21 – 33. <http://dx.doi.org/10.1109/2.86834>. – DOI 10.1109/2.86834. – ISSN 0018–9162

**Dreier 2006**

DREIER, R.: *Verteilte Ausführung hybrider System-Modelle auf heterogenen Rechnerplattformen*. Karlsruhe, Deutschland, Universität Fridericiana Karlsruhe, Diss., Mai 2006

**Dymola 2010**

DASSAULT SYSTÈMES (Hrsg.): *Multi-Engineering Modeling and Simulation - Dymola - CATIA - Dassault Systèmes*. Version: 2010. <http://www.3ds.com/products/catia/portfolio/dymola>, Abruf: 10.03.2010

**EUROSYSLIB 2010**

DASSAULT SYSTÈMES (Hrsg.): *Site EuroSyslib V2.18*. Version: 2010. <http://www.eurosyslib.com>, Abruf: 26.07.2010

**Extessy 2010**

EXTESSY AG (Hrsg.): *EXTESSY*. Version: 2010. <http://www.extessy.de>, Abruf: 08.07.2010

**Ferscha 1996**

FERSCHA, A.: Parallel and Distributed Simulation of Discrete Event Systems. In: ZOMAYA, A. (Hrsg.): *Parallel and Distributed Computing Handbook*, McGraw Hill Book Co, 1996. – ISBN 0–07–073020–2, S. 1003 – 1041

**Follen u. a. 2001**

FOLLEN, G. ; KIM, C. ; LOPEZ, I. ; SANG, J. ; TOWNSEND, S.: A CORBA-based Development Environment for Wrapping and Coupling Legacy Scientific Codes. In: *High-Performance Distributed Computing: 10th IEEE International Symposium*, I.E.E.E.Press, August 2001. – ISBN 978–0–769–51296–9. <http://dx.doi.org/10.1109/HPDC.2001.945173>. S. 22 – 31

**Försterling 2004**

FÖRSTERLING, S.: *Vergleichende Untersuchung von CO<sub>2</sub>-Verdichtern in Hinblick auf den Einsatz in mobilen Anwendungen*. Göttingen : Cuvillier, 2004. – ISBN 3–86537–080–2

**Fritzson 2004**

FRITZSON, P.: *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. 1. Wiley IEEE Press, 2004. – ISBN 978–0–471–47163–9

**Fujimoto 1990**

FUJIMOTO, R.: Parallel Discrete Event Simulation. In: *Commun. ACM* 33 (1990), Oktober, Nr. 10, S. 30 – 53. <http://dx.doi.org/10.1145/84537.84545>. – DOI 10.1145/84537.84545. – ISSN 0001–0782

**Fujimoto 1999**

FUJIMOTO, R.: Parallel and distributed simulation. In: *WSC '99: Proceedings of the 31st conference on Winter simulation*. New York, NY, USA : ACM, 1999. – ISBN 0–7803–5780–9. <http://dx.doi.org/10.1145/324138.324176>. S. 122 – 131

---

**Fujimoto 2000**

FUJIMOTO, R.: *Parallel and Distribution Simulation Systems*. New York, NY, USA : John Wiley & Sons, Inc., 2000. – ISBN 0471183830

**Fujimoto 2003**

FUJIMOTO, R.: Parallel simulation: distributed simulation systems. In: *WSC '03: Proceedings of the 35th conference on Winter simulation*, Winter Simulation Conference, 2003. – ISBN 0-7803-8132-7, S. 124 – 134

**Fujimoto u. a. 2007**

FUJIMOTO, R. ; HUNTER, M. ; SIRICHOKE, J. ; PALEKAR, M. ; KIM, H. ; SUH, W.: Ad Hoc Distributed Simulations. In: *PADS '07: Proceedings of the 21st International Workshop on Principles of Advanced and Distributed Simulation*. Washington, DC, USA : IEEE Computer Society, 2007. – ISBN 0-7695-2898-8. <http://dx.doi.org/10.1109/PADS.2007.10>. S. 15 – 24

**Fummy u. a. 2007**

FUMMY, F. ; LOGHI, M. ; PERBELLINI, G. ; PONCINO, M.: SystemC Co-Simulation for Core-Based Embedded Systems. In: *Design Automation for Embedded Systems 11* (2007), September, Nr. 2, S. 141 – 166. <http://dx.doi.org/10.1007/s10617-007-9006-7>. – DOI 10.1007/s10617-007-9006-7. – ISSN 0929-5585

**Gärtner u. a. 2000**

GÄRTNER, A. ; SAND, A. van d. ; RETTWEILER, P.: Auslegung von Fahrwerksystemen durch Co-Simulation von MKS- und Fluidsimulationssoftware. In: *9. Aachener Kolloquium Fahrzeug- und Motorentechnik*, 2000, 457 – 470

**Gear 1984**

GEAR, C.: Numerical Solution of Problems which may have High Frequency Component, 1984, S. 335 – 349

**Geimer u. a. 2006**

GEIMER, M. ; KRÜGER, T. ; LINSEL, P.: Co-Simulation, gekoppelte Simulation oder Simulator-kopplung? Ein Versuch der Begriffsvereinheitlichung. In: *O + P Zeitschrift für Fluidtechnik* 50 (2006), Nr. 11-12, S. 572 – 576. – ISSN 0341-2660

**Gelinas u. a. 1996**

GELINAS, B. ; DORMAN, K. ; MEDNICK, E.: LEAH: an introduction to behavioral abstraction and co-simulation using Perl and Verilog. In: *Verilog HDL Conference, IEEE International* (1996), S. 81 – 88. <http://dx.doi.org/10.1109/IVC.1996.496022>. – DOI 10.1109/IVC.1996.496022. – ISSN 1085-9403

**Gheorghe u. a. 2006**

GHEORGHE, L. ; BOUCHHIMA, F. ; NICOLESCU, G. ; BOUCHENEB, H.: Formal Definitions of Simulation Interfaces in a Continuous/Discrete Co-Simulation Tool. In: *Rapid System Prototyping, 2006. Seventeenth IEEE International Workshop on*, 2006. – ISSN 1074-6005. <http://dx.doi.org/10.1109/RSP.2006.18>. S. 186-192

**Ghosh 2002**

GHOSH, S.: The Role of Modeling and Asynchronous Distributed Simulation in Analyzing Complex

Systems of the Future. In: *Information Systems Frontiers* 4 (2002), Nr. 2, S. 161 – 177. <http://dx.doi.org/10.1023/A:1016098719029>. – DOI 10.1023/A:1016098719029. – ISSN 1387–3326

#### **GPS Visualizer 2011**

SCHNEIDER, A. (Hrsg.): *GPS Visualizer*. Version: 2011. <http://www.gpsvisualizer.com>, Abruf: 12.12.2010

#### **Gräber u. a. 2009**

GRÄBER, M. ; KOSOWSKI, K. ; RICHTER, C. ; TEGETHOFF, W.: Modeling of Heat Pumps with an Object-oriented Model Library for Thermodynamic Systems. In: TROCH, I. (Hrsg.) ; BREITENHECKER, F. (Hrsg.): *Proceedings MATHMOD 09 Vienna - Full Papers CD Volume*, 2009 (ARGESIM-Reports). – ISBN 978–3–901608–35–3

#### **Gustafson 1988**

GUSTAFSON, J.: Reevaluating Amdahl's law. In: *Commun. ACM* 31 (1988), Nr. 5, S. 532 – 533. <http://dx.doi.org/10.1145/42411.42415>. – DOI 10.1145/42411.42415. – ISSN 0001–0782

#### **Hackenberg 2006**

HACKENBERG, U.: Use of virtual techniques in Audi-product development – status quo and outlook. In: *Virtual Vehicle Creation 2006* ATZ, 2006

#### **Hager u. a. 2002**

HAGER, J. ; GUMPOLDSBERGER, T. ; REITBAUER, R.: Einsatz transienter thermischer Simulation von Nutzfahrzeugen zur Erhöhung von deren Wirtschaftlichkeit. In: DEUSSEN, N. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs III* Haus der Technik, Expert-Verlag, Juli 2002 (Haus der Technik Fachbuch 19). – ISBN 978–3–8169–2058–8, S. 39 – 60

#### **Hager u. a. 2003**

HAGER, J. ; STROH, C. ; REITBAUER, R.: Optimization strategies for the transient behaviour of the thermal system of commercial vehicles. In: *Vehicle Thermal Management Systems (VTMS 6)* SAE, Professional Engineering Publishing, Mai 2003. – ISBN 978–1–860–58418–3

#### **Hairer u. Wanner 1996**

HAIRER, E. ; WANNER, G.: *Solving Ordinary Differential Equations II – Stiff and Differential-Algebraic Problems*. 2. Düsseldorf : Springer, 1996. – ISBN 978–3–540–60452–5

#### **Hensen 1999**

HENSEN, J.: A comparison of coupled and de-coupled solutions for temperature and air flow in a building. In: *ASHRAE Transactions* Bd. 105 (1999), Nr. 2, S. 962 – 969

#### **Hessel u. Haase 2009**

HESEL, E. ; HAASE, J.: Aspekte der Transformationen zwischen VHDL-AMS- und Modelica-Modellen. (2009), März. ISBN 978–3–8167–7981–0

#### **Hesselbach 2003**

HESELBACH, J.: *Skript zur Vorlesung „Fertigungsautomatisierung 2“*. Institut für Werkzeugmaschinen und Fertigungstechnik, Technische Universität Braunschweig, 2003

#### **Hirano u. a. 1998**

HIRANO, S. ; YASU, Y. ; IGARASHI, H.: Performance Evaluation of Popular Distributed Object Technologies for Java. In: *Concurrency - Practice and Experience* 10 (1998), Nr. 11-13, S. 927–940

---

**IEEE 1076.1 1999**

IEEE Standard VHDL Analog and Mixed-Signal Extensions. In: *IEEE Std 1076.1-1999* (1999), März. ISBN 0-7381-1640-8

**IEEE 1278.1 1996**

IEEE Standard for Distributed Interactive Simulation - Application Protocols. In: *IEEE Std 1278.1-1995* (1996). <http://dx.doi.org/10.1109/IEEESTD.1996.80831>. – DOI 10.1109/IEEESTD.1996.80831

**IEEE 1516.3 2003**

IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP). In: *IEEE Std 1516.3-2003* (2003), S. 1 – 32

**Inokuty 1928**

INOKUTY, H.: Zeichnerisches Verfahren zum Auffinden des günstigsten Kondensatordruckes in Kohlensäurekältemaschinen. In: *Zeitschrift für die gesamte Kälte-Industrie* 9 (1928), S. 180 – 182

**Jackson 1991**

JACKSON, K.R.: A Survey of Parallel Numerical Methods for Initial Value Problems for Ordinary Differential Equations. In: *Magnetics, IEEE Transactions on* 27 (1991), September, Nr. 5, S. 3792 – 3797. <http://dx.doi.org/10.1109/20.104928>. – DOI 10.1109/20.104928. – ISSN 0018-9464

**Jones u. Fletcher 2005**

JONES, M. ; FLETCHER, D.: Thermal Performance Prediction of Front-End Heat-Exchange Modules. In: *Thermal Management Systems, Modelling and Components* Bd. SP-1945 Motor Industry Research Association, UK, 2005 (SAE World Congress 2005)

**Kanth 2010**

KANTH, D. ; EBERHARD, P. (Hrsg.): *Zur steifigkeits- und kopplungsbasierten Partitionierung mechatronischer Systeme*. 1. Aufl. Shaker, 2010 (Schriften aus dem Institut für Technische und Numerische Mechanik der Universität Stuttgart 17/2010). – ISBN 978-3-8322-9170-9

**Kim u. a. 2002**

KIM, Dohyung ; RHEE, Chan-Eun ; HA, Soonhoi: Combined data-driven and event-driven scheduling technique for fast distributed cosimulation. In: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 10 (2002), Oktober, Nr. 5, S. 672 – 678. <http://dx.doi.org/10.1109/TVLSI.2002.801572>. – DOI 10.1109/TVLSI.2002.801572. – ISSN 1063-8210

**Klumpp 2008**

In: KLUMPP, P.: *Steuertriebsentwicklung mit Simulation und Versuch*. 1. Vieweg + Teubner, 2008. – ISBN 978-3-8348-0345-0, S. 172 – 188

**Köhler u. a. 2008**

KÖHLER, C. ; MAYER, A. ; HERKERSDORF, A.: Determining the Fidelity of Hardware-In-the-Loop Simulation Coupling Systems. In: *Behavioral Modeling and Simulation Workshop, 2008. BMAS 2008. IEEE International*, 2008. – ISSN 978-1-4244-2896-0. <http://dx.doi.org/10.1109/BMAS.2008.4751232>. S. 13 – 18



**Köhler u. a. 1990**

KÖHLER, J. ; KÜHN, B. ; SONNEKALB, M. ; BEER, H.: Numerical Calculation of the Distribution of Temperature and Heat Flux in Buses Under the Influence of the Vehicle Air Conditioning System. In: *ASHRAE Transactions* 96 (1990), Nr. 3352, S. 432 – 446

**Kübler 2000**

KÜBLER, R.: *Fortschritts-Berichts VDI: Reihe 20, Rechnerunterstützte Verfahren*. Bd. 327: *Modulare Modellierung und Simulation mechatronischer Systeme*. Düsseldorf : VDI Verlag, 2000. – ISBN 3-18-332720-1

**Kübler u. Schiehlen 2000a**

KÜBLER, R. ; SCHIEHLEN, W.: Modular Simulation in Multibody System Dynamics. In: *Multibody System Dynamics* Bd. 4 (2000), August, Nr. 2, S. 107 – 127. <http://dx.doi.org/10.1023/A:1009810318420>. – DOI 10.1023/A:1009810318420. – ISSN 1384-5640

**Kübler u. Schiehlen 2000b**

KÜBLER, R. ; SCHIEHLEN, W.: Two Methods of Simulator Coupling. In: *Mathematical and Computer Modelling of Dynamical Systems* Bd. 6 (2000), Juni, Nr. 2, S. 93 – 113. [http://dx.doi.org/10.1076/1387-3954\(200006\)6:2;1-M;FT093](http://dx.doi.org/10.1076/1387-3954(200006)6:2;1-M;FT093). – DOI 10.1076/1387-3954(200006)6:2;1-M;FT093. – ISSN 1387-3954

**Lang u. a. 2006**

LANG, G. ; PUNTIGAM, W. ; PETUTSCHNIG, H. ; KITANOSKI, F.: Simulation des Aufwärmverhaltens von Verbrennungsmotor und Fahrzeug mittels Kopplung von Teilmodellen. In: BRILL, U. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs V* Haus der Technik, Expert-Verlag, 2006 (Haus der Technik Fachbuch 68). – ISBN 978-3-8169-2651-1, S. 38 – 53

**Le Marrec u. a. 1998**

LE MARREC, P. ; VALDERRAMA, C. A. ; HESSEL, F. ; JERRAYA, A. A. ; ATTIA, M. ; CAYROL, O.: Hardware, Software and Mechanical Cosimulation for Automotive Applications. In: *International Workshop on Rapid System Prototyping* IEEE Computer Society, 1998, S. 202 – 206

**Lee 2003**

LEE, E.: Overview of the Ptolemy Project. Version: Juli 2003. <http://ptolemy.eecs.berkeley.edu/publications/papers/03/overview/>, Abruf: 08.07.2010. 2003. – Forschungsbericht

**Lemke 2004**

LEMKE, N.: *Untersuchung zweistufiger Flüssigkeitskühler mit dem Kältemittel CO<sub>2</sub>*. Braunschweig, Deutschland, Technische Universität Braunschweig, Diss., Dezember 2004

**Lüdecke 2003**

LÜDECKE, A.: *Simulationsgestützte Verfahren für den Top-Down-Entwurf heterogener Systeme*. Duisburg, Germany, Universität Duisburg-Essen, Diss., Dezember 2003

**Lund u. a. 2005**

LUND, C. ; MAISTER, W. ; BEYER, B. ; WATTENBERG, M.: Bewertung der Kühlluftströmung im Motorraum mit 1D und 3D Verfahren. In: STEINBERG, P. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs IV* Haus der Technik, Expert-Verlag, Dezember 2005 (Haus der Technik Fachbuch 43). – ISBN 978-3-8169-2478-4, S. 29 – 57

---

**Lund u. a. 2008**

LUND, C. ; MAISTER, W. ; LANGE, C. ; BEYER, B.: Innovation durch Co-Simulation! In: BRILL, U. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs VI* Haus der Technik, Expert-Verlag, Juni 2008 (Haus der Technik Fachbuch 93). – ISBN 978-3-8169-2820-1, S. 163 – 187

**Maaßen u. Pischinger 2008**

*Kapitel 3.3.* In: MAASSEN, F. ; PISCHINGER, S.: *Virtuelle Motorenentwicklung*. 1. Vieweg + Teubner, 2008. – ISBN 978-3-8348-0345-0, S. 189 – 201

**Maué u. Kauf 2006**

MAUÉ, J. ; KAUF, F.: Thermischer Komfort in Fahrzeugkabinen. In: *Klimatisierung und Thermomanagement im KFZ*, IIR Deutschland, November 2006 (CTI Fachtagung)

**Michaelson u. Eiden 2009**

MICHAELSON, B. ; EIDEN, J.: HumanComfort Modelica-Library Thermal Comfort in Buildings and Mobile Applications. In: *Proceedings of the 7th International Modelica Conference, Como, Italy*. Linköping, Schweden : Linköping University Electronic Press, September 2009. – ISSN 1650-3686. <http://dx.doi.org/10.3384/ecp09430082>. S. 403 – 412

**Miller u. Thorpe 1995**

MILLER, D. ; THORPE, J.: SIMNET: the advent of simulator networking. In: *Proceedings of the IEEE* 83 (1995), August, Nr. 8, S. 1114 – 1123. <http://dx.doi.org/10.1109/5.400452>. – DOI 10.1109/5.400452. – ISSN 0018-9219

**Modelica 2009**

MODELICA ASSOCIATION (Hrsg.): *Modelica and the Modelica Association – Modelica Portal*. Version: 2009. <http://www.modelica.org>, Abruf: 14.10.2009

**Neudorfer u. a. 2006**

NEUDORFER, H. ; BINDER, A. ; WICKER, N.: Analyse von unterschiedlichen Fahrzyklen für den Einsatz von Elektrofahrzeugen. In: *e & i Elektrotechnik und Informationstechnik* 123 (2006), August, Nr. 7-8, S. 352 – 360. <http://dx.doi.org/10.1007/s00502-006-0363-x>. – DOI 10.1007/s00502-006-0363-x

**Newton u. Sangiovanni-Vincentelli 1983**

NEWTON, A.R. ; SANGIOVANNI-VINCENTELLI, A.L.: Relaxation-based electrical simulation. In: *Electron Devices, IEEE Transactions on* 30 (1983), September, Nr. 9, S. 1184 – 1207. – ISSN 0018-9383

**Nicol u. Heidelberger 1996**

NICOL, D. ; HEIDELBERGER, P.: Parallel Execution for Serial Simulators. In: *ACM Transactions on Modeling and Computer Simulation* 6 (1996), S. 210 – 242

**Nokia 2010**

NOKIA (Hrsg.): *Qt - A cross-platform application and UI framework*. Version: 2010. <http://qt.nokia.com>, Abruf: 02.05.2010

**Nötges u. a. 2007**

NÖTGES, T. ; HÖLEMANN, S. ; BAYER BOTERO, N. ; ABEL, D.: Objektorientierte Modellierung, Simulation und Regelung dynamischer Systeme am Beispiel eines Oxyfuel-Kraftwerksprozesses. In: *at – Automatisierungstechnik* 55 (2007), Mai, Nr. 5, S. 236 – 243. <http://dx.doi.org/10.1524/auto.2007.55.5.236>. – DOI 10.1524/auto.2007.55.5.236. – ISSN 0178-2312

**Oberschelp u. Vocking 2004**

OBERSCHELP, O. ; VOCKING, H.: Multirate simulation of mechatronic systems, 2004. <http://dx.doi.org/10.1109/ICMECH.2004.1364473>. S. 404 – 409

**Page u. a. 1999**

PAGE, E. ; NICOL, D. ; BALCI, O. ; FUJIMOTO, R. ; FISHWICK, P. ; L'ECUYER, P. ; SMITH, R.: Strategic directions in simulation research (panel). In: *WSC '99: Proceedings of the 31st conference on Winter simulation*. New York, NY, USA, 1999. – ISBN 0-7803-5780-9. <http://dx.doi.org/10.1145/324898.325314>. S. 1509 – 1520

**Peacock u. a. 1979**

PEACOCK, J. ; WONG, J. ; E., Manning.: Distributed Simulation Using a Network of Processors. In: *Computer Networks* 3 (1979), Nr. 1, S. 44 – 56

**Perschke u. Meinholf 2007**

PERSCHK, A. ; MEINHOLF, U.: Ein Modell zur hygrisch-thermischen Gebäudesimulation mit Hilfe der Kopplung von Zonen- und Feldmodell. In: *Bauphysik* 29 (2007), Nr. 1, S. 55 – 62. <http://dx.doi.org/10.1002/bapi.200710011>. – DOI 10.1002/bapi.200710011. – ISSN 0171-5445

**Petzold 1982**

PETZOLD, L.: A Description of DASSL: A Differential/Algebraic System Solver. In: *10th IMACS World Congress*, 1982, S. 65 – 68

**Pidd 2002**

PIDD, M.: Simulation Software and Model Reuse: A Polemic. In: *Simulation Conference, 2002. Proceedings of the Winter* Bd. 1, 2002. <http://dx.doi.org/10.1109/WSC.2002.1172959>. S. 772 – 775

**Puntigam 2008**

Kapitel 6.2. In: PUNTIGAM, W.: *Integrierte Virtuelle Gesamtfahrzeugsimulation ausgeführt am Beispiel des Thermischen Managements*. 1. Vieweg + Teubner, 2008. – ISBN 978-3-8348-0345-0, S. 358 – 378

**Puntigam u.a. 2004**

PUNTIGAM, W. ; BERNASCH, J. ; HAGER, J. ; REICH, F. ; RUMMEL, A.: Kopplung von 0D-, 1D-, und 3D CFD Simulationsprogrammen. In: *Automotive Engineering Partners* (2004), Nr. 2004-06, S. 48 – 51

**Puntigam u.a. 2005a**

PUNTIGAM, W. ; PETUTSCHNIG, H. ; HÖRMANN, T. ; ENNEMOSER, A. ; MOSHAMMER, T.: Future Trends for the Application of Numerical Tools in the Cooling System Development Process. In: *International Commercial Powertrain Conference* Bd. ICPC 2005 - 2.5, AVL List GmbH, April 2005, S. 86 – 102

---

**Puntigam u.a. 2005b**

PUNTIGAM, W. ; HÖRMANN, T. ; BERNASCH, J. ; SCHIERL, K. ; WIESLER, B. ; HAGER, J.: Thermisches Management im Fahrzeug durch Kopplung unterschiedlicher Simulationsmodelle. In: STEINBERG, P. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs IV* Haus der Technik, Expert-Verlag, Dezember 2005 (Haus der Technik Fachbuch 43). – ISBN 978-3-8169-2478-4, S. 141 – 157

**Puntigam u.a. 2006a**

PUNTIGAM, W. ; LANG, G. ; PETUTSCHNIG, H. ; ALMBAUER, R.: Instationäre Simulation des Thermischen Verhaltens von Fahrzeugen durch Koppelung von Teilmodellen am Beispiel des Motoraufwärmverhaltens. In: *Berechnung und Simulation im Fahrzeugbau* Bd. 1967.2, 2006 (VDI Berichte). – ISBN 3-18-091967-1, S. 963 – 1008

**Puntigam u.a. 2006b**

PUNTIGAM, W. ; BALIC, J. ; ALMBAUER, R. ; HAGER, J.: Transient Co-Simulation of Comprehensive Vehicle Models by Time Dependent Coupling. In: *SAE 2006 Transactions Journal of Passenger Cars: Mechanical Systems*. 2006 ( 2006-01-1604). – ISBN 978-0-7680-1838-7, S. 1516 – 1525

**Raab u. a. 2008**

RAAB, M. ; SCHULZE, T. ; STRASSBURGER, S.: Erfahrungen aus der Anwendung von HLA-basierter verteilter Simulation im Nutzfahrzeugbereich. In: RABE, M. (Hrsg.): *13. ASIM-Fachtagung Simulatio in Produktion und Logistik*, Fraunhofer IRB Verlag, Oktober 2008, S. 499 –508

**Rat der EG 1970**

RAT DER EUROPÄISCHEN GEMEINSCHAFTEN: *Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedsstaaten über Maßnahmen gegen die Verunreinigung der Luft durch Emissionen von Kraftfahrzeugen*. 1970 ( 70/220/EWG)

**Regner u. a. 2002**

REGNER, G. ; LOIBNER, E. ; KRAMMER, J.: Analysis of Transient Drive Cycles Using Cruise-Boost Co-Simulation Techniques. In: *SAE 2002 World Congress*. United States : SAE International, März 2002

**Richert u. a. 2005**

RICHERT, F. ; BOLLIG, A. ; ABEL, D.: Modellbildung eines Dieselmotors unter Matlab und Dymola zur Reglerentwicklung mit modernen Simulationswerkzeugen. In: GÜMANN, C. (Hrsg.) ; WOLTER, T.-M. (Hrsg.) ; Haus der Technik (Veranst.): *Simulation und Test in der Funktions- und Softwareentwicklung für die Automobilelektronik* Haus der Technik, Expert-Verlag, März 2005 (Haus der Technik Fachbuch 45). – ISBN 978-3-8169-2491-3, S. 29 – 38

**Riederer u. a. 2009**

RIEDERER, P. ; KEILHOLZ, W. ; DUCREUX, V.: Coupling of Trnsys with Simulink – a Method to Automatically Export and Use Trnsys Models within Simulink and vice versa. In: *Eleventh International IBPSA Conference* IBPSA, 2009, S. 1628 – 1633

**Robinson u. a. 2004**

ROBINSON, S. ; NANCE, R. ; PAUL, R. ; PIDD, M. ; TAYLOR, S.: Simulation model reuse: definitions, benefits and obstacles. In: *Simulation Modelling Practice and Theory* 12 (2004), Nr. 7-8, S. 479 –

494. <http://dx.doi.org/10.1016/j.simpat.2003.11.006>. – DOI 10.1016/j.simpat.2003.11.006.  
– ISSN 1569–190X. – Simulation in Operational Research

**Samuel u. a. 2002**

SAMUEL, S. ; AUSTIN, L. ; MORREY, D.: Automotive test drive cycles for emission measurement and real-world emission levels - A review. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 216 (2002), Nr. 7, S. 555 – 564. <http://dx.doi.org/10.1243/095440702760178587>. – DOI 10.1243/095440702760178587

**Sargent 1988**

SARGENT, R.: A tutorial on validation and verification of simulation models. In: *WSC '88: Proceedings of the 20th conference on Winter simulation*. New York, NY, USA : ACM, Dezember 1988. – ISBN 0–911801–42–1. <http://dx.doi.org/10.1145/318123.318150>. S. 33 – 39

**Sargent 2005**

SARGENT, R.: Verification and validation of simulation models. In: *WSC '05: Proceedings of the 37th conference on Winter simulation*. Piscataway, NJ, USA : IEEE Press, 2005. – ISBN 0–7803–9519–0, S. 130 – 143

**Sargent 2007**

SARGENT, R.: Verification and validation of simulation models. In: *WSC '07: Proceedings of the 39th conference on Winter simulation*. Piscataway, NJ, USA : IEEE Press, 2007. – ISBN 1–4244–1306–0, S. 124 – 137

**Sargent 2008**

SARGENT, R.: Verification and validation of simulation models. In: *WSC '08: Proceedings of the 40th Conference on Winter Simulation*. Piscataway, NJ, USA : IEEE Press, 2008. – ISBN 978–1–4244–2708–6, S. 157 – 169

**Sargent u. a. 2006**

SARGENT, R. ; NANCE, R. ; OVERSTREET, C. ; ROBINSON, S. ; TALBOT, J.: The Simulation Project Life-Cycle: Models and Realities. In: *WSC '06: Proceedings of the 38th conference on Winter simulation*. Piscataway, NJ, USA : IEEE Press, 2006. – ISBN 1–4244–0501–7, S. 863 – 871

**Schmidt 1992**

SCHMIDT, D.: Systems programming with C++ wrappers: encapsulating IPC services with object-oriented interfaces. In: *C++ Report* 4 (1992), Oktober, Nr. 8, S. 50 – 54. – ISSN 1040–6042

**Schmidt u. a. 1995**

SCHMIDT, D. ; HARRISON, T. ; AL-SHAER, E.: Object-oriented components for high-speed network programming. In: *COOTS'95: Proceedings of the USENIX Conference on Object-Oriented Technologies on USENIX Conference on Object-Oriented Technologies (COOTS)*. Berkeley, CA, USA : USENIX Association, 1995

**Schneider u. a. 2005**

SCHNEIDER, F. ; BUNZL, A. ; HOFHAUS, J. ; BRAUN, M. ; LIMPERICH, D. ; CÄSAR, R. ; ARNTZ, K. ; SCHRÖTER, A. ; SPECHT, B. ; MÖNKEDIEK, T.: Entwicklung und Einführung eines einheitlichen Kältekreislauf-Simulationsprogramms. In: SCHLENZ, D. (Hrsg.) ; Haus der Technik (Veranst.): *PKW-Klimatisierung IV* Haus der Technik, Expert-Verlag, Oktober 2005 (Haus der Technik Fachbuch 11). – ISBN 978–3–8169–2554–5, S. 185 – 195

---

**Schröder u. a. 2002**

SCHRÖDER, K. ; WAGNER, S. ; ELLINGER, M.: Gekoppelte Simulation der Klimaanlage und Fahrgastzelle unter Berücksichtigung variierender Randbedingungszyklen. In: SCHLENZ, D. (Hrsg.) ; Haus der Technik (Veranst.): *PKW-Klimatisierung II* Haus der Technik, Expert-Verlag, Juli 2002 (Haus der Technik Fachbuch 11). – ISBN 978-3-8169-2033-5, S. 210 – 223

**Seiffert u. Rainer 2008**

SEIFFERT, U. (Hrsg.) ; RAINER, G. (Hrsg.): *Virtuelle Produktentstehung für Fahrzeug und Antrieb im Kfz*. 1. Vieweg + Teubner, 2008. – ISBN 978-3-8348-0345-0

**Shi 1996**

SHI, Y.: *Reevaluating Amdahl's Law and Gustafson's Law*. <http://www.cis.temple.edu/~shi/docs/amdahl/amdahl.html>. Version: Oktober 1996, Abruf: 23.06.2009

**Simscape 2010**

THE MATHWORKS, INC. (Hrsg.): *Simscape 3 User's Guide*. The MathWorks, Inc., 2010. [http://www.mathworks.com/access/helpdesk/help/pdf\\_doc/physmod/simscape/simscape\\_ug.pdf](http://www.mathworks.com/access/helpdesk/help/pdf_doc/physmod/simscape/simscape_ug.pdf)

**Straßburger u.a. 2001**

STRASSBURGER, S.: *Distributed Simulation Based on the High Level Architecture in Civilian Application Domains*. Magdeburg, Germany, Otto-von-Guericke-Universität Magdeburg, Diss., April 2001

**Straßburger u.a. 2003**

STRASSBURGER, S. ; SCHMIDGALL, G. ; HASSIS, S.: Distributed Manufacturing Simulation as an Enabling Technology for the Digital Factory. In: *Journal of Advanced Manufacturing Systems* 2 (2003), Juni, Nr. 1, S. 111 – 126

**Taylor u. a. 2002**

TAYLOR, S. ; BRUZZONE, A. ; FUJIMOTO, R. ; GAN, B. ; STRASSBURGER, S. ; PAUL, R.: Distributed Simulation and Industry: Potentials and Pitfalls. In: *Proceedings of the 2002 Winter Simulation Conference*. Los Alamitos, CA, USA : IEEE Computer Society, 2002. – ISBN 0-7803-7614-5. <http://dx.doi.org/10.1109/WSC.2002.1172850>. S. 688 – 694

**Thomke 2003**

THOMKE, S.: *Experimentation Matters: Unlocking the Potential of New Technologies for Innovation*. Mcgraw-Hill Professional, 2003. – ISBN 978-1578517503

**Tiller 2001**

TILLER, M.: *The Springer International Series in Engineering and Computer Science*. Bd. 615: *Introduction to Physical Modeling with Modelica*. 1. Springer, 2001. – ISBN 978-0-7923-7367-4

**Tonchev u. a. 2007**

TONCHEV, A. ; HIRSCHBERG, W. ; JAGSCH, S.: Investigation of the Thermal Vehicle Brake Behavior During the Vehicle's Development Phase by Co-Simulation. In: *25th Annual Brake Colloquium and Exhibition*. United States : SAE International, Oktober 2007

**Trčka 2008**

TRČKA, M.: *Co-simulation for Performance Prediction of Innovative Integrated Mechanical Energy*

*Syystems in Buildings*. Eindhoven, Niederlande : Technische Universiteit Eindhoven, 2008. – ISBN 978-90-386-1366-6

**Trčka u. a. 2007**

TRČKA, M. ; WETTER, M. ; HENSEN, J.: Comparison of co-simulation approaches for building and HVAC-R system simulation. In: INTERNATIONAL BUILDING PERFORMANCE SIMULATION ASSOCIATION (Hrsg.): *10th IBPSA Conference, Beijing, China*, 2007, S. 1418 – 1425

**Valderrama u. a. 1998**

VALDERRAMA, C. ; NACABAL, F. ; PAULAN, P. ; JERRAYA, A.: Automatic VHDL-C Interface Generation for Distributed Cosimulation: Application to Large Design Examples. In: *Design Automation for Embedded Systems* Bd. 3 (1998), Juni, Nr. 2 - 3, S. 199 – 217. <http://dx.doi.org/10.1023/A:1008898525388>. – DOI 10.1023/A:1008898525388. – ISSN 0929-5585

**Weatherly u. a. 1993**

WEATHERLY, R. ; WILSON, A. ; GRIFFIN, S.: ALSP - Theory, Experience, and Future Directions. In: *Simulation Conference Proceedings, 1992. Winter*, 1993. – ISBN 0-7803-1381-X, S. 1068 – 1072

**Weigand u. a. 2008**

WEIGAND, B. ; KÖHLER, J. ; WOLFERSDORF, J. von: *Thermodynamik kompakt*. 1. Springer, Berlin, 2008. – ISBN 978-3-540-71865-9

**Wenzel u. a. 2007**

WENZEL, S. ; WEISS, M. ; COLLISI-BÖHMER, S. ; PITSCH, H. ; ROSE, O.: *Qualitätskriterien für die Simulation in Produktion und Logistik: Planung und Durchführung von Simulationsstudien*. 1. Springer, Berlin, 2007. – ISBN 978-3-540-35272-3

**Wilcox u. a. 2000**

WILCOX, P. ; BURGER, A. ; HOARE, P.: Advanced distributed simulation: a review of developments and their implication for data collection and analysis. In: *Simulation Practice and Theory* 8 (2000), Nr. 3-4, 201 – 231. [http://dx.doi.org/10.1016/S0928-4869\(00\)00023-9](http://dx.doi.org/10.1016/S0928-4869(00)00023-9). – DOI 10.1016/S0928-4869(00)00023-9. – ISSN 0928-4869

**Wünsche u.a. 1997a**

WÜNSCHE, S. ; CLAUSS, C. ; SCHWARZ, P. ; WINKLER, F.: Electro-thermal circuit simulation using simulator coupling. In: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 5 (1997), September, Nr. 3, S. 277-282. <http://dx.doi.org/10.1109/92.609870>. – DOI 10.1109/92.609870. – ISSN 1063-8210

**Yahiaoui u. a. 2003**

YAHIAOUI, A. ; HENSEN, J. ; SOETHOUT, L.: Integration of control and building performance simulation software by run-time coupling. In: INTERNATIONAL BUILDING PERFORMANCE SIMULATION ASSOCIATION (Hrsg.): *8th IBPSA Conference, Eindhoven, Netherlands*, 2003, S. 1435 – 1442

**Zhai 2003**

ZHAI, Z.: *Developing an integrated building design tool by coupling building energy simulation and computational fluid dynamics programs*. Cambridge, Massachusetts, USA, Massachusetts Institute of Technology, Diss., September 2003





## B. Eigene Veröffentlichungen

### Kossel u.a. 2006

KOSSEL, R. ; TEGETHOFF, W. ; BODMANN, M. ; LEMKE, N.: Simulation of Complex Systems using Modelica and Tool Coupling. In: *Proceedings of the 5th International Modelica Conference, Vienna, Austria*, Modelica Association, September 2006, 485 - 490

### Kossel u.a. 2008

KOSSEL, R. ; FÖRSTERLING, S. ; TEGETHOFF, W.: Einsatz hybrider Simulationstechnik für die Bewertung mobiler Heiz- und Kühlkonzepte. In: BRILL, U. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs VI* Haus der Technik, Expert-Verlag, Juni 2008 (Haus der Technik Fachbuch 93). – ISBN 978-3-8169-2820-1, S. 150 – 162

### Kossel u.a. 2009a

KOSSEL, R. ; STRUPP, N. C. ; TEGETHOFF, W.: Effects of Tool Coupling on Transient Simulation of a Mobile Air-Conditioning Cycle. In: *Proceedings of the 7th International Modelica Conference, Como, Italy*. Linköping, Schweden : Linköping University Electronic Press, September 2009. – ISSN 1650-3686. <http://dx.doi.org/10.3384/ecp09430064>. 318 – 325

### Kossel u.a. 2009b

KOSSEL, R. ; CORREIA, C. ; LOEFFLER, M. ; BODMANN, M. ; TEGETHOFF, W.: Verteilte Systemsimulation mit TISC. In: *ASIM-Workshop 2009 in Dresden mit integrierter DASS'2009*, 2009

### Kossel u.a. 2010

KOSSEL, R. ; LOEFFLER, M. ; BODMANN, M. ; TEGETHOFF, W.: Verteilte Simulation von Energiesystemen im Fahrzeug. In: *Berechnung und Simulation im Fahrzeugbau 2010*, VDI-Verlag, November 2010

### Kossel u.a. 2011

KOSSEL, R.M. ; LOEFFLER, M. ; STRUPP, N.C. ; TEGETHOFF, W.J.: Distributed energy system simulation of a vehicle. In: *VTMS 10 – Vehicle Thermal Management Systems*, Institution of Mechanical Engineers, Mai 2011

### Lemke u.a. 2011

LEMKE, N. ; STRUPP, N.C. ; KOSSEL, R.: Elektrofahrzeug-Klimatisierung unter Berücksichtigung relevanter Klima- und Lastbedingungen. In: MAGER, R. (Hrsg.) ; Haus der Technik (Veranst.): *PKW-Klimatisierung VII* Haus der Technik, Expert-Verlag, November 2011 (Haus der Technik Fachbuch 124). – ISBN 978-3-8169-3113-3

### Loeffler u. a. 2006

LOEFFLER, M. ; HUHN, M. ; RICHTER, C. ; KOSSEL, R.: Modelica CVD - A Tool for Visualizing

---

the Structure of Modelica Libraries. In: *Proceedings of the 5th International Modelica Conference, Vienna, Austria*, Modelica Association, September 2006, 55 - 62

**Strupp u.a. 2009**

STRUPP, N.C. ; LEMKE, N. ; TEGETHOFF, W. ; KOSSEL, R. ; KÖHLER, J.: Betriebsverhalten von Automobilverflüssigern. In: HOFHAUS, J. (Hrsg.) ; Haus der Technik (Veranst.): *PKW-Klimatisierung VI* Haus der Technik, Expert-Verlag, November 2009 (Haus der Technik Fachbuch 107). – ISBN 978-3-8169-2934-5

**Strupp u.a. 2010a**

STRUPP, N.C. ; KÖHLER, J. ; TEGETHOFF, W.J. ; LEMKE, N.C. ; KOSSEL, R.M.: Energy Efficient Future Automotive Condenser System. In: *Conference on Thermal and Environmental Issues in Energy Systems* The American Society of Mechanical Engineers (ASME), 2010

**Strupp u.a. 2010b**

STRUPP, N.C. ; KÖHLER, J. ; LEMKE, N. ; TEGETHOFF, W. ; KOSSEL, R.: Energy Efficient Future Automotive Condenser Systems. In: *2010 International Symposium on Next-Generation Air Conditioning and Refrigeration Technology* New Energy and Industrial Technology Development Organization (NEDO), 2010

**Strupp u.a. 2010c**

STRUPP, N.C. ; KÖHLER, J. ; KOSSEL, R. ; TEGETHOFF, W.: Senkung des Kraftstoffverbrauchs durch Optimierung der Leerlaufklimatisierung eines PKW mittels Hybridkühlung. In: *Deutsche Kälte-Klima-Tagung 2010* Deutscher Kälte- und Klimatechnischer Verein e.V., DKV, 2010

**Tegethoff u.a. 2006a**

TEGETHOFF, W. ; CORREIA, C. ; KOSSEL, R. ; BODMANN, M. ; LEMKE, N. ; KÖHLER, J.: Co-Simulation und Sprach-Standardisierung am Beispiel des Wärmemanagements. In: BRILL, U. (Hrsg.) ; Haus der Technik (Veranst.): *Wärmemanagement des Kraftfahrzeugs V* Haus der Technik, Expert-Verlag, Juni 2006 (Haus der Technik Fachbuch 68). – ISBN 978-3-8169-2651-1, S. 231 – 242

**Tegethoff u.a. 2006b**

TEGETHOFF, W. ; KOSSEL, R. ; HABIG, R.: Koppelung von Software-Werkzeugen zur Simulation der thermischen Behaglichkeit. In: *Klimatisierung und Thermomanagement im KFZ*, IIR Deutschland, November 2006 (CTI Fachtagung)

**Tegethoff u.a. 2007**

TEGETHOFF, W. ; HORST, T. ; RICHTER, C. ; KOSSEL, R. ; BODMANN, M. ; LEMKE, N.: Modeling and Simulation of Compressors in Mobile Air-Conditioning Systems. In: *Proceedings of the 6th EUROSIM Congress on Modelling and Simulation*. Ljubljana, Slovenia, September 2007